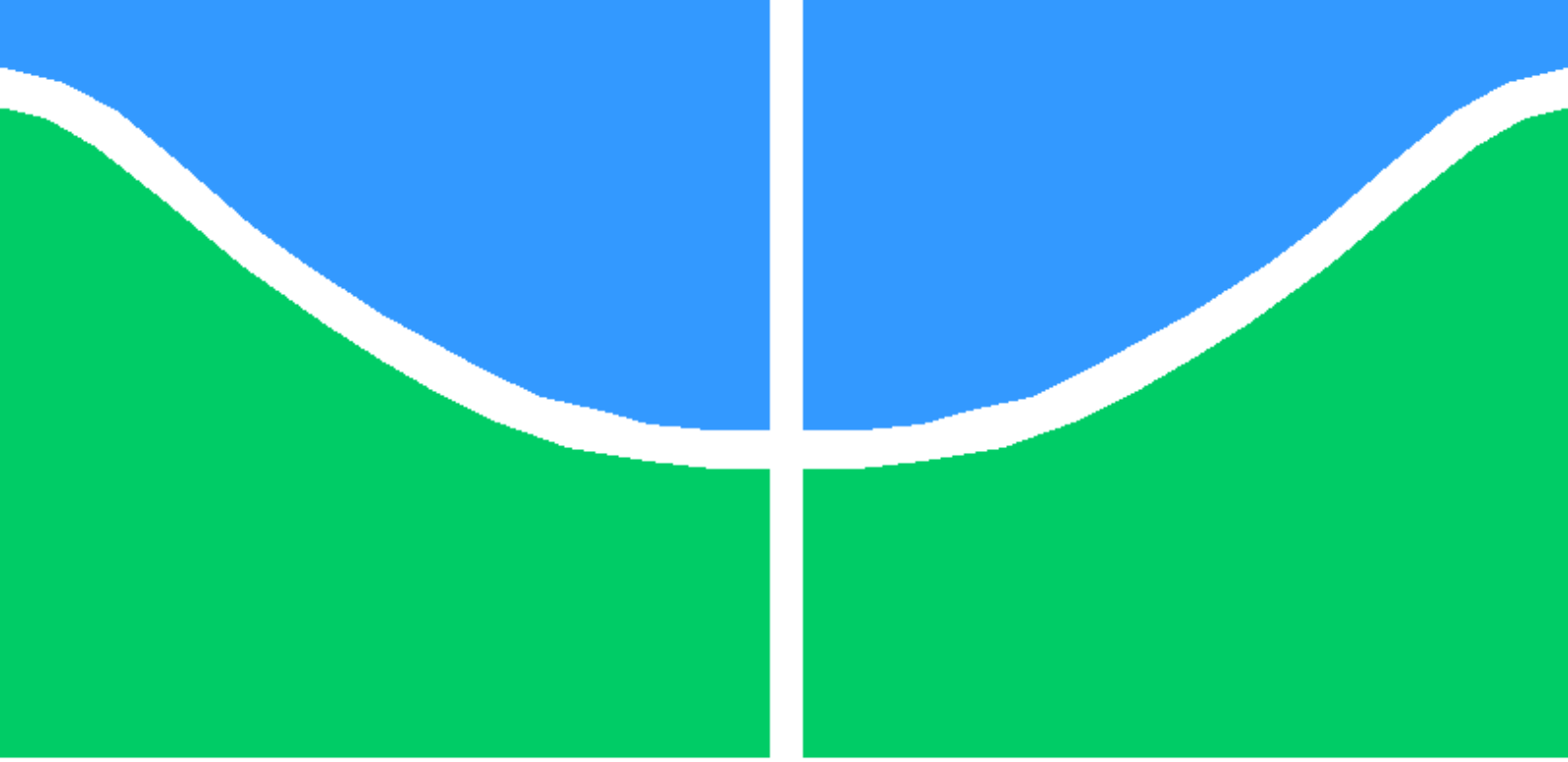


TRABALHO DE GRADUAÇÃO

EduCRAS
Proposição de um modelo de monitoramento distribuído do eduroam no Brasil

Calebe Souza Reis

Brasília, Dezembro de 2018

UNIVERSIDADE DE BRASÍLIA

FACULDADE DE TECNOLOGIA

UNIVERSIDADE DE BRASILIA
Faculdade de Tecnologia

TRABALHO DE GRADUAÇÃO

EduCRAS
Proposição de um modelo de monitoramento distribuído do eduroam no Brasil

Calebe Souza Reis

Relatório submetido ao Departamento de Engenharia Elétrica como requisito parcial para obtenção do grau de Engenheiro de Redes de Comunicação

Banca Examinadora

Prof. Georges Daniel Amvamé Nzé., Dr., _____
ENE/UnB
Orientador

Prof. Rafael Timóteo de Sousa Jr., Dr., _____
ENE/UnB
Examinador Interno

Jean Carlo Faustino, Dr., RNP _____
Examinador Externo

Um especialista é um homem que cometeu todos os erros que podem ser cometidos em um campo muito restrito.

Traduzido, Niels Bohr

Agradecimentos

Agradeço primeiramente a Deus, por nunca ter tirado a minha vida do Seu controle. Aos meus pais, por me apoiarem e me incentivarem em tudo. Ao meu irmão e minha namorada, por me suportarem e sempre acreditarem em mim. Aos meus amigos e familiares, por estarem sempre orando e torcendo pelo meu sucesso. Aos meus colegas da RNP, que acreditaram e me ajudaram para o sucesso deste trabalho.

Calebe Souza Reis

RESUMO

O serviço eduroam faz parte de um projeto de colaboração mundial entre as redes de ensino e pesquisa que permite a alunos e membros de instituições o acesso à internet de forma rápida e segura em qualquer instiuição do mundo que faça parte da rede eduroam com as credenciais da instiuição de origem. No Brasil, o serviço é mantido pela RNP. Desde que foi implementado, o eduroam cresce a cada ano e, a partir de um determinado momento, ficou complicado monitorar toda a utilização da rede no Brasil. Este trabalho apresenta uma solução para o monitoramento do eduroam utilizando *Machine Learning* para detectar comportamentos anômalos e, a partir disso, criar alertas para os administradores do serviço.

ABSTRACT

Eduroam is part of a global collaboration project between the research and education networks that allows students and members of these networks to access the internet quickly and safely in any institution in the world with eduroam using their local credentials. In Brazil, eduroam is maintained by RNP. since it was implemented, eduroam has increased every year and, in a certain moment, it becomes complicated to monitor all the Brazilian uses of the service eduroam. This project presents a monitoring solution for eduroam using *Machine Learning* to detect anomalies and, from this, generate alerts to the service administrators.

SUMÁRIO

LISTA DE FIGURAS

LISTA DE TABELAS

LISTA DE ABREVIATURAS

Acrônimos

ACK	Acknowledgement
AES	Advanced Encryption Standard
AP	Access Point
CAFe	Comunidade Acadêmica Federada
CCMP	Counter Mode CBC MAC protoco
DAP	Directory Access Protocol
DIT	Directory Information Tree
EAP	Extensible Authentication Protocol
ELK	Elasticserach e Logstash e Kibana
ESS	Extended Service Set
FTP	File Transfer Protocol
IEEE	Institute of Electrical and Electronics Engineers
IETF	Internet Engineering Task Force
GB	Gigabyte
IdP	Provedor de Identidade
IP	Internet Protocol
JSON	JavaScript Object Notation
LDAP	Lightweight Directory Access Protocol
MB	Megabyte
MIC	Message Integrity Code
NREN	National Research and Education Network
RADIUS	Remote Authentication Dial In User Service
RAM	Random Access Memory

Acrônimos

RNP	Rede Nacional de Ensino e Pesquisa
RSN	Robust Security Network
RTS	Request To Send
SIEM	Security Information and Event Management
SP	Provedor de Serviço
SSH	Secure Shell
SSID	Service Set Identifier
SSL	Secure Sockets Layer
TB	Terabyte
TCP	Transmission Control Protocol
TERENA	Trans-European Research andEducation Networking Association
TKIP	Temporal Key Integrity Protocol
TLS	Transport Layer Security
TSN	Transitional Security Network
UDP	User Datagram Protocol
VM	Virtual Machine
VPN	Virtual Private Network
WAYF	Where Are You From
WEP	Wired Equivalent Privacy
WLAN	Wireless Local Area Network
WNIC	Wireless Network Interface Controller
WPA	Wi-Fi Protected Access
WPS	Wi-Fi Protected Setup

Capítulo 1

Introdução

A rede eduroam, do inglês *education roaming*, é uma rede sem fio com o principal objetivo de proporcionar a usuários de redes acadêmicas acesso seguro e gratuito à internet em qualquer lugar do mundo onde haja um *Access Point* configurado. Desenvolvida em 2002 pela Comunidade de Pesquisa e Educação Europeia, foi inicialmente utilizada com a participação de seis países europeus, Holanda, Alemanha, Finlândia, Portugal, Croácia e Reino Unido. Esse modo de utilização de *Wi-Fi* chegou ao Brasil em 2011, quando foi realizado, pela RNP (Rede Nacional de Ensino e Pesquisa), um projeto piloto com as Universidades Federal Fluminense, Federal do Rio de Janeiro e Federal do Mato Grosso do Sul. Ao final deste piloto, o projeto Eduroam-br tornou-se um serviço do portfólio da RNP.

Todo usuário cadastrado em uma rede acadêmica federada, como a CAFe (Comunidade Acadêmica Federada) no Brasil, que faz parte do eduroam tem direito a acessar à rede, bastando apenas fornecer suas credenciais de acesso na hora da autenticação. Isso porque as instituições contam com um servidor RADIUS (*Remote Authentication Dial In User Service*) e uma base de dados LDAP (*Lightweight Directory Access Protocol*), que são as ferramentas necessárias para que haja a conexão com o servidor RADIUS central, mantido pela NREN (*National Research and Education Network*) de cada país, RNP no Brasil, possibilitando o acesso.

Segundo a GÉANT (GÉANT, 2018), mantenedora dos direitos administrativos, o eduroam alcançou, em novembro de 2018, a marca de 101 países cadastrados. No Brasil, existem 126 instituições, com 2540 *Access Points* utilizando o serviço eduroam, que tem como principal ferramenta dessa rede o processo de *roaming*, onde acontece a conexão de usuários em instituições diferentes da sua instituição de origem. A Figura 1.1 mostra como funciona esse processo.

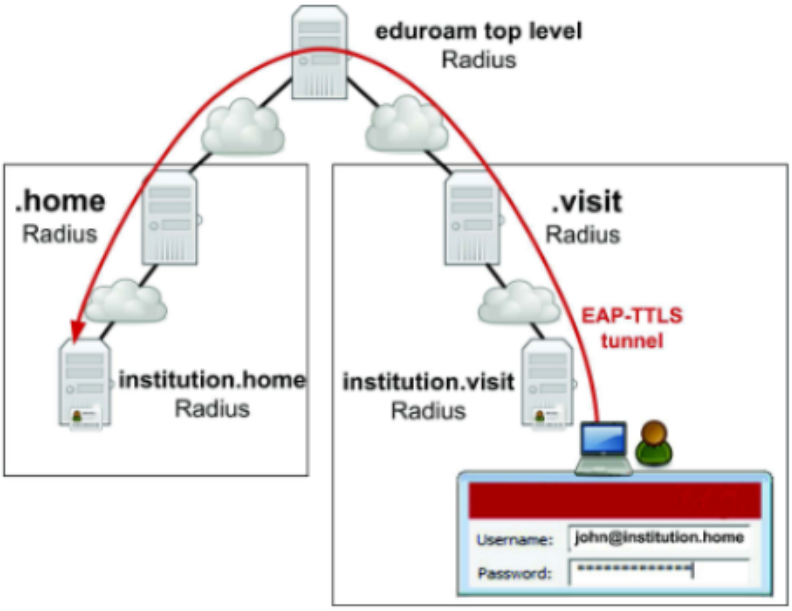


Figura 1.1: Esquema de utilização do processo de roaming. Fonte: (IFSC, 2012)

1.1 Definição do Problema

Com o constante crescimento da utilização do eduroam desde a sua criação, em 2002, a operação e o monitoramento desse serviço têm se tornado cada vez mais difícil de ser realizada por seres humanos, principalmente devido ao custo de pessoal relacionado a esse trabalho. Identificar, através de varreduras constantes, instituições que estão com problemas na utilização, ou até mesmo instituições que não estão fazendo uso do serviço é uma tarefa que não tem sido realizada desde o início da operação do serviço eduroam no Brasil.

Com isso, surge uma nova possiblidade para manter o bom nível de atuação da rede, que é a utilização de aprendizado de máquina para auxiliar a gerência do serviço. Com o uso desta tecnologia, podemos avançar ao nível de identificarmos possíveis indisponibilidades nas instituições clientes antes mesmo que a própria instituição a identifique, através do monitoramento em tempo real do servidor RADIUS da federação.

1.2 Objetivo

O Objetivo Geral deste projeto é implementar um sistema de monitoramento do serviço de roaming do eduroam no Brasil, através dos logs do servidor da RNP. Para isso, será realizada uma prova de conceito com a utilização de uma ferramenta própria para gerenciamento de logs.

1.3 Objetivos Específicos

Para alcançar este objetivo, os seguintes objetivos específicos são propostos:

- Definir qual a melhor ferramenta para o projeto;
- Entender como funciona os logs do eduroam;
- Analisar quais métricas podem ser medidas;
- Exportar os logs para o servidor do monitoramento;
- Criar dashboards com análises sobre a utilização do serviço;
- Implementar o *Machine Learning* para análise comportamental;
- Gerar alertas sobre as anomalias detectadas.

1.4 Estrutura do Trabalho

O restante deste trabalho é organizado da seguinte forma. O capítulo 2 consiste em uma apresentação dos conceitos básicos que compõem o trabalho. O Capítulo 3 mostra como foi feita a implementação do ambiente de monitoramento, detalhando desde a ferramenta escolhida até a arquitetura implementada no projeto. O Capítulo 4 contém as análises dos resultados obtidos, com uma discussão a respeito da performance da ferramenta no monitoramento dos logs do servidor e sobre a robustez da solução. O Capítulo 5 contém a discussão final sobre o projeto e conclui o trabalho, apresentando possibilidades para possíveis melhoras e trabalhos futuros.

Capítulo 2

Fundamentação Teórica

Neste capítulo serão apresentados os principais conceitos utilizados no desenvolvimento desse trabalho.

2.1 Eduroam

O education roaming, eduroam, surgiu de uma iniciativa da Trans-European Research and Education Networking Association (TERENA) de fornecer um serviço de internet sem fio seguro para membros de comunidades acadêmicas de ensino e pesquisa. O eduroam permite que os usuários tenham acesso à internet de forma rápida e segura dentro de suas instituições e também quando visitam outras insituições que fazem parte da federação eduroam.

Para melhor visualização do funcionamento, suponhamos que um professor da Universidade A, o Marcos, irá palestrar em um evento na Universidade B. Ao chegar lá, para acessar à internet ele recebe uma credencial com login 'marcos.a' e senha 'marcos123' e, assim, acessa à rede com tranquilidade. Na semana seguinte, o professor Marcos viaja para participar de um evento na Universidade C e, assim como na B, recebe credenciais de acesso à internet, porém diferentes da anterior. Na mesma semana, Marcos também visita a Universidade D e, mais uma vez, recebe credenciais diferentes. Um tempo depois, ele precisa retornar à Universidade B e, como recebeu diversas credenciais, não se lembra mais quais são as certas para aquela instituição e acaba precisando pedir novas credenciais.

Neste exemplo vimos quão trabalhosa é toda essa situação de acesso à internet para visitantes, para o usuário e, principalmente, para a instituição, que precisa gerir toda essa logística de criação e distribuição de credenciais, além de políticas sobre a utilização como tempo de validade das credencias, por exemplo. O eduroam serve como uma alternativa eficiente para essa gestão de acesso das instituições, fazendo com as credenciais de um usuário na sua instituição de origem sejam válidas em qualquer outra instituição que faça parte da federação.

A Figura 2.1 mostra como funciona essa troca de informações segura para que o usuário consiga acessar à rede sem fio com as credenciais da sua instituição de origem.

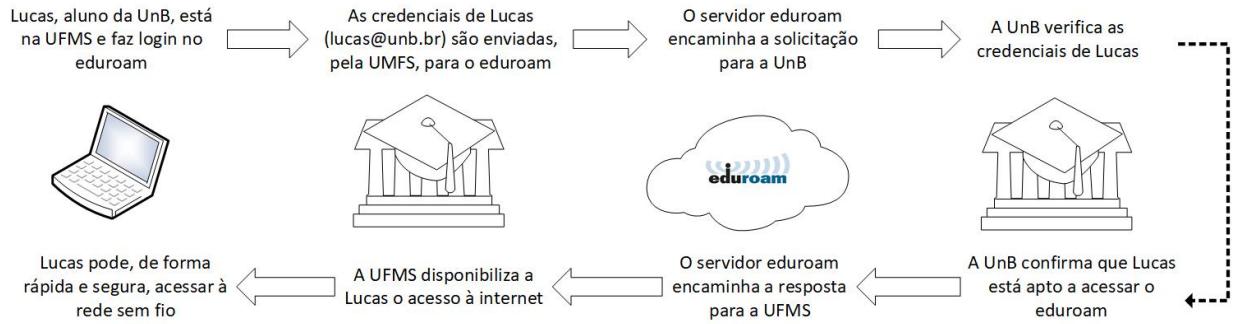


Figura 2.1: Modelo de funcionamento Eduroam

2.1.1 Roaming

A esse processo visto na seção anterior, damos o nome de *roaming*. Este processo é o principal objeto de estudo deste trabalho, pois atua como a principal função do serviço eduroam, que é tornar transparente e seguro ao usuário da comunidade federada o acesso à rede em uma insituição diferente da sua instituição de origem. O *roaming* torna possível a autenticação de um usuário com uma única senha em qualquer instituição disponível.

O eduroam utiliza uma estrutura hierárquica de três níveis de servidores de autenticação, como pode ser visto na Figura 2.2. No primeiro nível estão as instituições participantes de uma mesma federação. No segundo nível estão as federações pertencentes a uma mesma confederação. E, no terceiro nível estão as confederações.

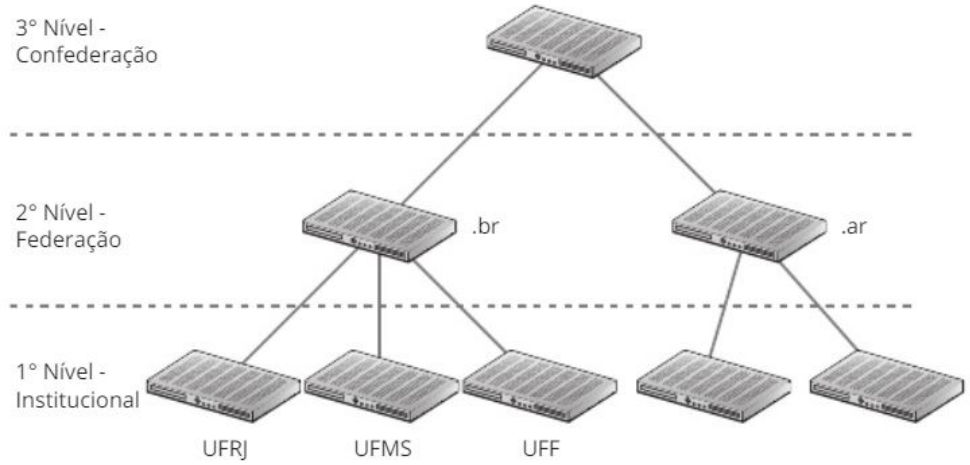


Figura 2.2: Hierarquia de servidores do eduroam em nível continental. Fonte: (SAADE, 2013)

Retomando ao exemplo de utilização do eduroam que vimos na Figura 1.1, temos um novo exemplo do processo de roaming dentro dessa visão de hierárquia de servidores na Figura 2.3. Nesse exemplo, um usuário da instituição B (universidade.br) está tentando acessar ao eduroam na instituição A (instituto.br). Quando o usuário faz a requisição, a intituição A checa o domínio relacionado àquele usuário e, ao verificar que não é compatível com o domínio pela qual ela responde, a requisição é encaminhada para o servidor de nível superior, no caso o da federação (mas

poderia subir até o da confederação, caso necessário). O servidor da federação, então, identifica a qual instituição aquele usuário está relacionado e encaminha a requisição, no caso para a instituição B que, por sua vez, verifica as credenciais do usuário e retorna para a federação se estas são válidas. Por último, a federação encaminha a resposta da instituição B para a instituição A, para que esta possa liberar ou não o acesso à rede para aquele usuário.

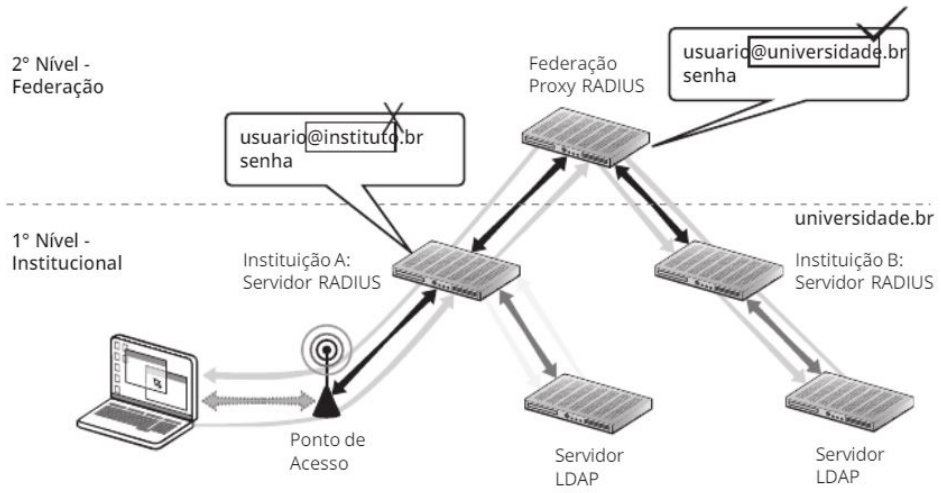


Figura 2.3: Exemplo de roaming na visão hierárquica. Fonte: (SAADE, 2013)

2.2 Tecnologias Utilizadas no eduroam

Como vimos na sessão anterior, o serviço eduroam é baseado em uma arquitetura de servidores de autenticação, em que os pedidos de autenticação dos usuários são tratados em suas instituições de origem. Quando um usuário está em sua instituição de origem a solicitação é tratada pelo servidor de autenticação local. Quando, porém, ele está em uma instituição diferente da sua, acontece o processo visto na Figura 2.1.

O eduroam utiliza o padrão internacional *Remote Authentication Dial In User Service* (RADIUS), publicado pelo *Internet Engineering Task Force* (IETF). Quanto à infraestrutura de rede necessária para oferecer o serviço, são utilizados pontos de acesso sem fio IEEE 802.11, sendo apoiado pelo padrão 802.1X e também no padrão 802.11i para prover mecanismos de segurança no acesso. Para a autenticação dos usuários, as informações devem ser armazenadas em diretórios utilizando bases LDAP (*Lightweigh Directory Access Protocol*). As instituições são responsáveis por garantir a credibilidade das credenciais dos seus usuários, portanto, é necessário que haja uma relação de confiança entre as instituições parceiras e, para isso, o eduroam utiliza o conceito de federação. No Brasil, existe a Comunidade Acadêmica Federada (CAFe) e, para que uma instituição possa oferecer o serviço eduroam, esta deve ser um Provedor de Identidade (IdP) cadastrado na CAFe.

2.2.1 CAFe

A Comunidade Acadêmica Federada (CAFe) é um serviço de gestão de identidade que utiliza sua integração de suas bases de dados para reunir instituições de ensino e pesquisa brasileiras. Neste serviço, é utilizado o modelo Single Sign-on para que o usuário, através de uma única conta, possa acessar os serviços da sua própria instituição e, também, os oferecidos por outras organizações participantes da federação. A utilização de uma comunidade federada como essa traz consigo diversos benefícios para os usuários e as instituições. Se, pela ótica do usuário, ele só precisa memorizar um usuário e senha, para a instituição isso traz a vantagem de otimizar espaço em seus servidores de base de dados.

O funcionamento da CAFe é baseado na relação dos usuários com as instituições, que atuam de duas formas: como Provedor de Identidade (IdP) e Provedor de Serviço (SP). Os IdP's são responsáveis por manter as informações sobre usuários e por sua autenticação, enquanto os SP's são os ofertantes dos serviços utilizados. A interação entre os dois é constituída de uma relação de confiança, pois é preciso garantir que os dados fornecidos são verdadeiros e que serão usados para o que foi combinado.

Quando um usuário acessa a um determinado SP e tenta fazer login, ele é redirecionado a uma página, WAYF (Where Are You From?), que lhe mostrará uma lista com os IdP's disponíveis. Assim, ele escolhe sua instituição de origem e fornece suas credenciais. Essas credenciais são enviadas para o IdP indicado e, caso sejam válidas, o SP recebe a resposta e autoriza o usuário a acessar àquele serviço. A figura 2.4 nos traz, de forma mais detalhada, todo esse processo de autenticação. Há também, uma outra utilidade desse serviço que, uma vez logado em um SP, o usuário não precisa, em um determinado período de tempo, fazer todo o processo de login novamente, caso queira acessar a outro serviço.

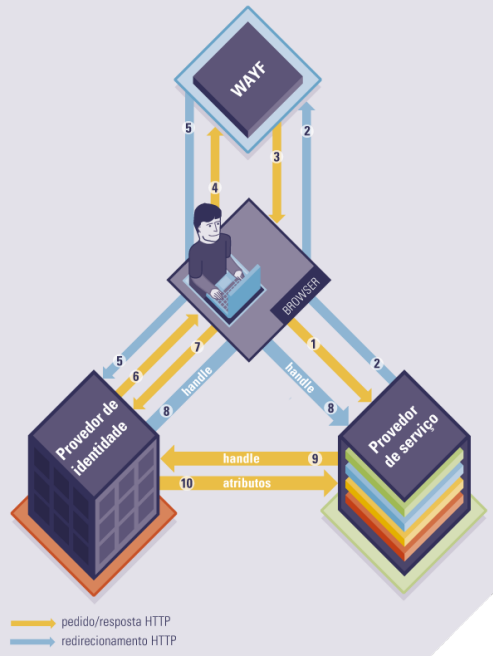


Figura 2.4: Modelo de autenticação na CAFe. Fonte: (RNP, 2018)

2.2.2 Arquitetura 802.11

Lançado em 1997, o padrão IEEE 802.11 (IEEE, 1999), é um conjunto de especificações a respeito das WLANs (*Wireless Local Area Networks*). O IEEE atualiza este padrão regularmente, lançando versões com melhorias em diversas áreas, como aspectos de segurança, qualidade de serviço e até na camada física. Algumas dessas atualizações, no entanto, não trazem uma redefinição do padrão, mas apenas alguns incrementos, como é o caso da versão 802.11i, que especifica um novo método para melhorar a segurança, e a versão 802.11r, que apresenta melhorias no processo de *roaming*. A Tabela 2.1 apresenta algumas versões deste padrão e algumas características.

Tabela 2.1: Padrões IEEE 802.11 (IEEE, 1999).

	802.11a	802.11b	802.11g	802.11n	802.11ac
Lançamento	1999	1999	2003	2009	2013
Taxa de dados	54mbps	11mbps	54mpbs	300mbps	1300mbps
Frequência	5GHz	2,4GHz	2,4GHz	2,4/5GHz	5GHz
Modulação	OFDM	DSSS	OFDM	MIMO-OFDM	MIMO-OFDM

2.2.2.1 Conexão

A conexão realizada entre um ponto de acesso e o dispositivo solicitante é realizada em duas etapas, a associação e a autenticação (IEEE, 2004). A etapa de associação, detalhada na Figura 2.5, tem seu início no recebimento de datagramas *beacon* pelo dispositivo solicitante. Após a escolha da rede desejada, uma requisição de autententicação é enviada ao ponto de acesso, que responde com um pacote de resposta de autenticação, liberando o dispositivo para enviar uma requisição de associação, e, por último, envia a resposta de associação. Quando a associação é bem sucedida, dá-se início à etapa de autenticação que, se não for através da utilização de um protocolo de segurança, já pode iniciar o envio de dados.

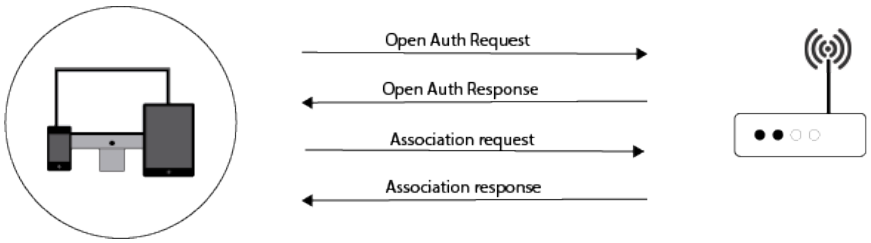


Figura 2.5: Processo de conexão em uma rede WPA *Personal*

2.2.2.2 Segurança em Redes Sem Fio

Tendo em vista a comunicação em meio aberto e compartilhado, o monitoramento de redes sem fio acaba sendo mais fácil se comparado a redes cabeadas. Para tentar diminuir esse risco de espionagem, foi criado o protocolo WEP, visando prover integridade, confidencialidade e controle

de acesso às redes sem fio (IEEE, 2004). Porém, o protocolo WEP possuía diversas vulnerabilidades e, por isso, o IEEE desenvolveu um padrão com maior segurança, o IEEE 802.11i, publicado em 2004.

Inicialmente, o 802.11i definia um padrão que requeria certos recursos inviáveis à época aos dispositivos, o *Robust Security Network* (RSN), o que forçou o IEEE criar um novo padrão, que seria compatível com os dispositivos utilizados na época e, assim, surgiu o *Transitional Security Network* (TSN).

O TSN, também chamado de WPA, conta com uma segurança maior que a presente no WEP e compartilha a mesma arquitetura de autenticação e gerenciamento de chaves com o RSN. O protocolo de criptografia utilizado em uma rede WPA é o *Temporal Key Integrity Protocol* (TKIP). Este protocolo, quando comparado ao WEP, apresenta melhorias como chaves maiores, de 40 bits utilizados no WEP para 128, e a implementação de um algoritmo que incorpora a proteção contra ataques de *replay*, conhecido como *Michael*. Também é possível observar o suporte a mecanismos de autenticação como o 802.1X ou o EAP.

Já o RSN, que também ficou conhecido como WPA2 (KAMMERSTETTER et al., 2016), é considerado mais seguro que o WPA, porém requer a utilização de hardwares mais potentes para a realização de múltiplas operações de criptografia. O protocolo utilizado pelo WPA2 é o *Counter Mode CBC MAC protocol* (CCMP), baseado no *Advanced Encryption* (AES), que ainda não teve sua criptografia quebrada e, por isso, é considerado seguro ainda hoje.

Como foi dito, o WPA e o WPA2 compartilham a mesma arquitetura. Eles podem operar baseados em dois modos: o *preshared key* (PSK) e o 802.1X. O PSK, chamado de WPA *personal* é utilizado em ambientes menores, como casas e pequenas empresas, enquanto o 802.1X, chamado de WPA *enterprise*, foi desenvolvido para ser utlizado em ambientes maiores, como universidades e grandes empresas. O WPA *enterprise*, a partir do *Extensible Authentication Protocol* (EAP), permite a integração com outro protocolos de autenticação, através da conexão dos pontos de acesso a servidores de autenticação, como é o caso do eduroam. No PSK, a autenticação é realizada através de uma chave configurada no ponto de acesso e compartilhada com os dispositivos solicitantes.

O WPA *personal* possui autenticação baseada no *4-way handshake*, que possui segurança vulnerável a técnicas de descoberta de senha, como a técnica de dicionário, por exemplo. Essa vulnerabilidade é um dos pontos que permitem que o WPA *personal* seja considerado inseguro, principalmente quando comparado ao WPA *enterprise*.

O *4-way handshake*, representado na Figura 2.6, é efetuado a partir da troca de quatro mensagens entre o dispositivo solicitante e o ponto de acesso. A primeira delas é enviada pelo ponto de acesso ao dispositivo, contendo um número aleatório conhecido como *ANonce*. A segunda mensagem é enviada pelo socilitante, contendo outro número aletório, o *SNonce*, e uma mensagem *Message Integrity Code* (MIC), que visa garantir a integridade do pacote e que mostra que o cliente possui a chave correta. Já a terceira mensagem, enviada pelo AP, comprova que o ponto de acesso também conhece a chave e garante a autenticação. Por último, a quarta mensagem, enviada pelo dispositivo, informa ao ponto de acesso que será instalada a chave e que o tráfego criptografado será iniciado.

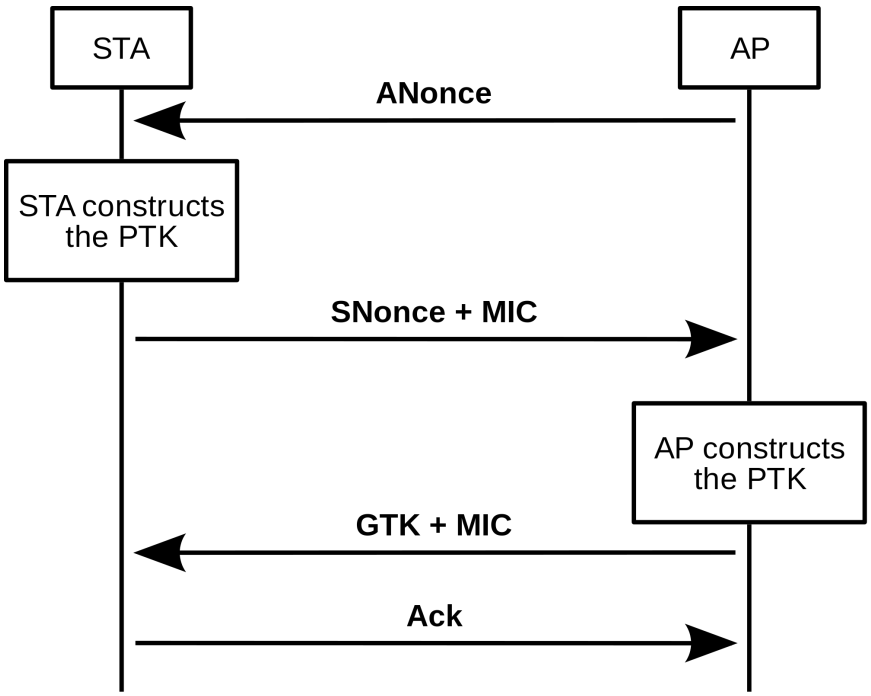


Figura 2.6: *4-way handshake*

As chaves utilizadas no *4-way handshake* são geradas a partir da utilização da função PBKDF2 (Password-Based Key Derivation Function 2), que tem como objetivo dificultar a quebra de senha através de métodos de força bruta. As entradas que precisam ser passadas para a função são o SSID, a PSK, o tamanho da chave e a quantidade de iterações que devem ser realizadas. Essa função dificulta a quebra de senha de duas formas, a primeira é a utilização do SSID como parâmetro, o que impossibilita a criação de tabelas de chaves pré calculadas a partir da PSK, e a segunda é a utilização, por padrão, de 4096 iterações, aumentando o tempo de geração da chave criptográfica.

2.2.3 RADIUS

O Remote Authentication Dial In User Service (RADIUS) é um padrão IETF, publicado inicialmente pela RFC 2058 (IETF, 1997) e, posteriormente, incrementada pelas RFCs 2138 , 2865 e 2866. O RADIUS é utilizado para prover serviço de autenticação centralizada em diversos tipos de redes de acesso, como *Virtual Private Networks* (VPNs), redes cabeadas e redes sem fio.

O serviço adota o modelo denominado como cliente-servidor, onde o cliente RADIUS é responsável por obter a informação sobre o cliente final e repassá-la para o servidor RADIUS, que, por sua vez, recebe e verifica as credenciais do usuário. O cliente atua, tipicamente, como um intermediário entre o dispositivo do usuário final e o servidor. O tipo de rede utilizada determina o tipo de cliente utilizado. Em redes sem fio, como é o caso do eduroam, o cliente é do tipo *Network Acess Server* (NAS) e quem atua como tal é o ponto de acesso da rede. Já o servidor RADIUS é responsável por receber os pedidos de conexão, autenticar o usuário e repassar ao cliente as informações necessárias para este dar acesso à rede ao usuário. As credenciais passadas pelo usuário e recebidas pelo servidor, por intermédio do cliente, são verificadas em bases de dados que podem

estar armazenadas localmente, em forma de arquivo texto, ou em elementos externos, como bases relacionais (SQL) e hierárquicas (LDAP), como é o caso do eduroam.

Quando utilizado em redes locais, o padrão RADIUS pode ser utilizado com o padrão *Extensible Authentication Protocol* (EAP). Como vimos anteriormente, o padrão 802.1X descreve como o EAP é encapsulado nos padrões IEEE 802. A Figura 2.7 mostra o esquema de encaminhamento de autenticação de um cliente final até o servidor RADIUS, onde o suplicante faz a requisição ao NAS, no caso do eduroam um ponto de acesso, e este encaminha a requisição ao servidor RADIUS, que fará a verificação e enviará a resposta ao NAS a cada solicitação recebida.

Figura 2.7: Encaminhamento de autenticação de um cliente até o servidor RADIUS. Fonte: (SAADE, 2013)

O protocolo RADIUS utiliza o protocolo de transporte UDP para enviar suas mensagens encapsuladas através das portas 1812 e 1813. As mensagens são encapsuladas no campo de dados UDP, deixando indicado a porta de destino usada pelo servidor. A Figura 2.8 traz uma amostra do formato de uma mensagem do protocolo RADIUS. Ela é formada por cinco campos principais, são eles:

- *Code*: esse campo é formado por um *byte* e identifica o tipo da mensagem RADIUS.
- *Identifier*: esse campo, ocupado por um *byte* auxilia nos pedidos e respostas do servidor RADIUS, identificando possíveis mensagens duplicadas.
- *Length*: Ocupa dois *bytes* e define o tamanho do pacote.
- *Authenticator*: esse campo ocupa 16 *bytes* e é utlizado para autenticar a resposta requerida pelo servidor RADIUS ou esconder a senha. O *byte* mais significativo é o transmitido inicialmente.
- *Attribute*: o campo de atributos é responsável por autenticar, autorizar, informare configurar detalhes para pedidos e respostas entre cliente e servidor.

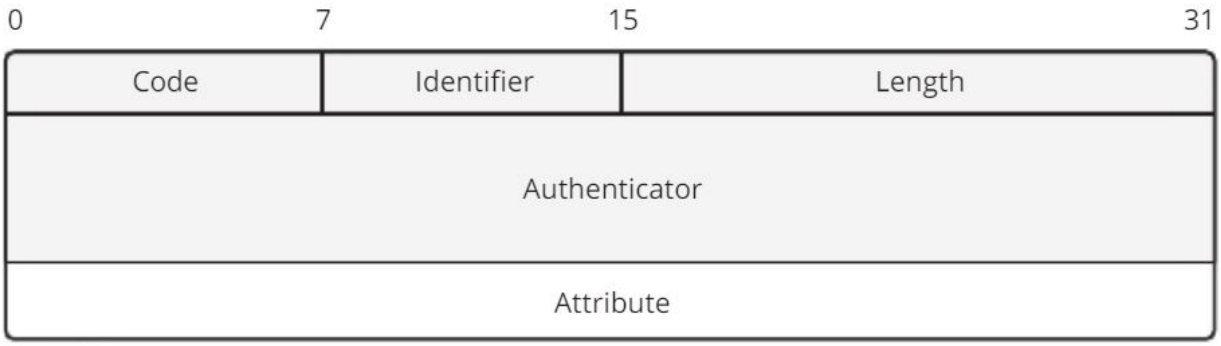


Figura 2.8: Formato da mensagem RADIUS. Fonte: (SAADE, 2013)

2.2.4 LDAP

Um diretório é uma base de dados que armazena informações sobre pessoas, grupos, instituições, entidades e serviços. Os diretórios se diferenciam dos Banco de Dados, como SQL por exemplo, pelo fato de serem mais focados em armazenar uma grande quantidade de dados e facilitar a busca e navegação por esses dados. Isso faz dos diretórios sistemas otimizados para leitura. As operações de ecrita e atualização são suportadas, porém são menos comuns.

O *Lightweigh Directory Acess Protocol* (LDAP), ou Protocolo Leve de Acesso a Diretório, surgiu como uma melhoria ao *Directory Access Protocol* (DAP), criado para ser um protocolo de acesso ao padrão de diretórios X.500. A União Internacional de telecomunicações especificou o padrão X.500 nos anos 80, juntamente com o protocolo de acesso DAP. O DAP era baseado no modelo *Open Systems Interconnection* (OSI). O LDAP veio então como uma alternativa mais leve ao DAP e, principalmente, como uma tecnologia nova, por ser baseado no modelo TCP/IP, que começava a substituir as redes baseadas no modelo OSI. A RFC 4510 especifica o padrão LDAP no IETF.

O LDAP, assim como o RADIUS, segue o modelo cliente-servidor. No LDAP, porém, há uma árvore de informações do diretório (*Directory Information Tree - DIT*), onde um ou mais servidores contêm os dados que perfazem essa árvore. Um exemplo de DIT é o mostrado na Figura 2.9, que detalha a visualização de uma árvore baseada nos nomes de domínios da internet.

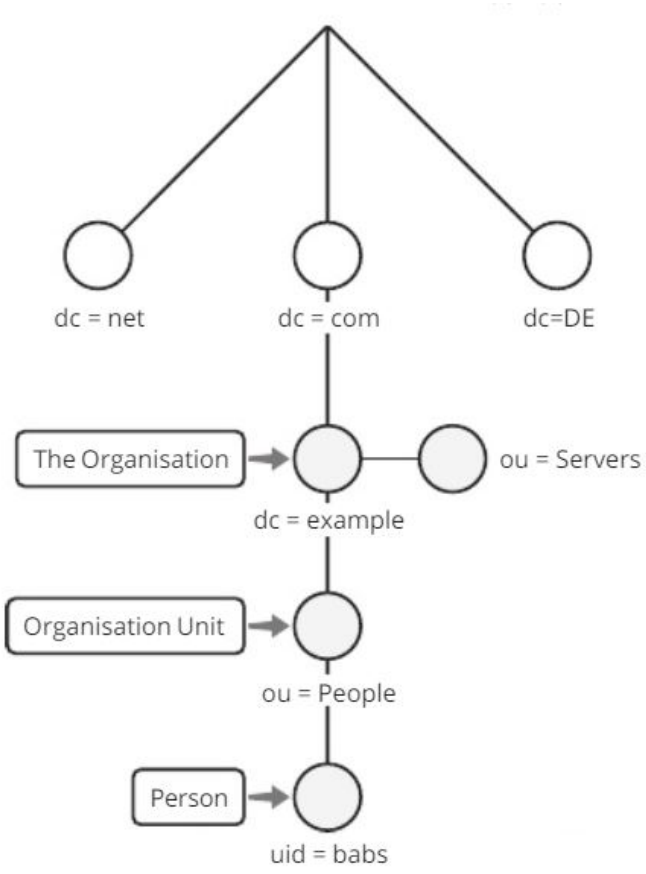


Figura 2.9: Estrutura de uma Árvore de Informações de Diretório - DIT. Fonte: (SAADE, 2013)

O cliente LDAP faz a comunicação com os servidores enviando requisições com o protocolo LDAP. O protocolo LDAP utiliza o transporte confiável oferecido pelo TCP e o servidor LDAP, por padrão, escuta na porta 389. No LDAP, o cliente pode enviar diversas requisições, independente de ter recebido resposta para as anteriores, e o servdor pode responder a essas requisições em qualquer ordem.

O cliente LDAP tem as seguintes opções de requisições que podem ser feitas:

- StartTLS: iniciar uma conexão criptografada
- Bind: autenticar e especificar a versão do LDAP utilizada
- Unbind: encerrar uma conexão
- Search: fazer uma busca no diretório
- Compare: verificar se uma entrada possui determinado valor
- Abandon: abandonar uma requisição
- Add/Delete/Modify/Move/Modify DN: adicionar, remover, modificar, mover ou renomear uma entrada.

2.2.5 RadSec

Quando o protocolo RADIUS é executado junto com TLS dá-se o nome de RadSec (RADIUS over TLS). Este modo de execução do RADIUS ainda não é um padrão, mas uma proposta de padrão, tratada pela IETF como rascunho. O primeiro rascunho sobre o RadSec está na RFC 6614, de 2012. A principal mudança implementada pelo RadSec é a alteração no protocolo de comunicação utilizado, pois o que é tradicionalmente utilizado pelo RADIUS é o UDP, porém o RadSec propõe utilizar o TCP e proteger a comunicação pelo protocolo criptográfico TLS. A ideia de se fazer essa mudança é justamente aumentar a segurança e a confiabilidade da comunicação entre o autenticador e o servidor de autenticação, ou entre os servidores de autenticação, quando for um sistema de autenticação hierárquico, como o que é utilizado no eduroam.

O padrão RADIUS apresenta algumas deficiências que podem ser perigosas para a segurança de um sistema tão grande como no caso do eduroam. A autenticação de servidores e clientes RADIUS é baseada nos endereços IP desses elementos e em senhas pré-compartilhadas compartilhadas nesses dispositivos. Essas senhas não são configuradas diretamente, mas são utilizadas para o cálculo de resumos criptográficos gerados pelo algoritmo MD5. Acontece que esse algoritmo já é, comprovadamente, um algoritmo fraco. Entre outras vulnerabilidades desse algoritmo, é possível encontrar duas mensagens produzindo um mesmo resumo criptográfico (*hash*). Além disso, o RADIUS apresenta outra falha de segurança considerada grave, pois, em uma comunicação, são protegidas algumas partes, como a senha dos usuários, deixando as outras partes da mensagem transmitida sem qualquer proteção.

Como método obrigatório de autenticação, o padrão RadSec utiliza o certificado X.509, que permite a criação de uma comunicação segura e mutuamente autenticada. Esse modelo de segurança é considerado forte, uma vez que provê um alto nível de criptografia, onde toda a mensagem é criptografada antes de ser enviada ao servidor RADIUS, e também no caminho inverso. Além disso, a utilização do protocolo TCP ao invés do UDP traz uma maior confiabilidade na transmissão, funcionalidade nativa do TCP que é de suma importância em sistemas de comunicação como o utilizado no eduroam.

Hoje, há duas duas implementações possíveis para o RadSec. A primeira delas é conhecida como *Radiator*, que é bastante utilizada, porém é paga. A segunda implementação é o *radsecproxy*, versão grauita do RadSec, compatível com o FreeRADIUS. No eduroam, é utilizado o *radsecproxy*. A Figura 2.10 mostra que o *radsecproxy* funciona com UDP em comunicações internas, mas só envia e recee mensagens externas via TCP/TLS.

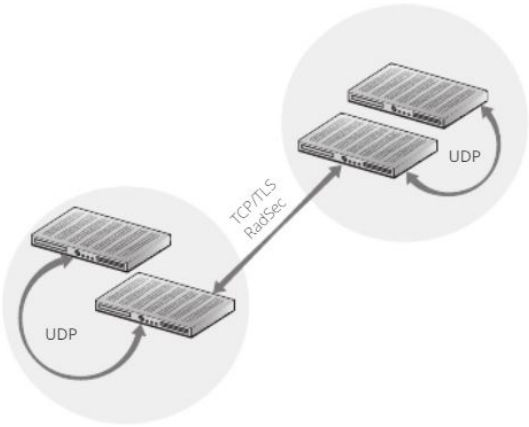


Figura 2.10: Comunicação interna do FreeRADIUS (UDP) e o encaminhamento pelo RadSec (TCP/TLS). Fonte: (SAADE, 2013)

2.3 Machine Learning

O *Machine Learning*, ou Aprendizado de Máquina, em português, é um ramo da Inteligência Artificial que se baseia na ideia de que sistemas podem aprender com dados, identificando padrões e tomando decisões com o mínimo de intervenção humana possível. Esse, portanto, é um método de análise de dados que automatiza a construção de modelos analíticos.

O termo Inteligência Artificial foi criado em 1956, quando Allen Newell e Herbert Simon desenvolveram o *logic theoristic*, programa capaz de demonstrar teoremas não triviais da lógica matemática, baseando-se na utilização de sistemas simbólicos e na introdução de heurísticias. Com o passar dos anos, esse novo metódo de computação foi tornando-se conhecido. A *Defense Advanced Research Projects Agency* (DARPA) dos EUA, completou, nos anos 1970, um projeto de mapeamento de ruas. A DARPA criou também, em 2003, assistentes pessoais inteligentes, como a Siri ou Cortana que conhecemos hoje. A I.A. se baseia na ideia de ensinar máquinas a fazerem tarefas humanas, como fazer contas matemáticas extremamente rápido. o *Machine Learning*, por sua vez, é uma parte da I.A. que trata de treinar as máquinas a como aprender sobre alguma coisa, procurando por padrões nos dados que a capacite de tomar decisões.

Segundo a empresa (SAS, 2018), o Aprendizado de Máquina tem um aspecto iterativo de extrema importância, pois, quando os métodos são expostos a novos dados, eles se adaptam independentemente. Esses métodos aprendem com computações anteriores para produzir resultados e decisões confiáveis. Um exemplo de utilização do Aprendizado de Máquina nos dias de hoje são os carros autônomos da Google, que recebem uma enorme quantidade de dados por segundo e, a partir disso, é capaz de identificar o que pode ou não fazer enquanto dirige e até se preparar para possíveis decisões erradas de motoristas próximos.

O *Machine Learning* se baseia em diferentes métodos, sendo esses definidos de acordo com o modelo de utilização a ser seguido. Abaixo, podemos ver alguns dos métodos mais conhecidos,

sendo os dois primeiros os mais utilizados.

- Aprendizado supervisionado: Os algoritmos são treinados por exemplo "rotulados", onde há saídas determinadas para determinadas entradas. Nesse tipo de método, o algoritmo recebe um conjunto de entradas junto com as saídas corretas correspondentes, e aprende ao comparar a saída real com as saídas corretas esperadas para, assim, identifcar erros.
- Aprendizado não supervisionado: Este método é utilizado quando com dados que não possuem rótulos históricos, ou seja, a saída correta não é conhecida do algoritmo, o que faz com que o sistema precise descobrir o que está nos dados. O objetivo de se utilizar o Aprendizado de Máquina assim é dar ao sistema o objetivo de encontrar uma estrutura dentro da amostra de dados e, assim, determinar o padrão.
- Aprendizado semi-supervisionado: É utilizado para aplicações parecidas com as do aprendizado supervisionado. Porém, este método manipula tanto dados rotulados, quanto dados não-rotulados. Muitas vezes é alimentado com uma quantidade menor de dados rotulados e outra maior de dados não-rotulados, devido ao fato de que os rotulados são mais caros e demandam mais esforço para serem adquiridos.
- Aprendizado por esforço: Este é o método normalmente usado em jogos, navegação e robótica, onde o algoritmo descobre quais ações rendem mais recompensas através de testes do tipo tentativa e erro. Possui três componentes principais: o agente (que vai aprender e tomar decisões), o ambiente (as condições com as quais o agente interage) e as ações (o que o agente faz). o objetivo é que o agente possa tomar ações que gerem a maior recompensa dentro de um determinado período de tempo.

Como vimos no início deste seção, o Aprendizado de Máquina é uma parte da Inteligência Artificial. Outras partes muito utilizadas nos dias de hoje são a Mineração de Dados e o *Deep Learning*. Esses conceitos são facilmente confundidos por aqueles que não tem conhecimento profundo da área. Embora esses métodos tenham o mesmo objetivo, de extrair padrões e relações de dados a fim de tomar decisões, cada um possui abordagens e capacidades diferentes.

A mineração de dados tem como objetivo principal a utilização de métodos diferentes para extrair *insights* dos dados, procurando por padrões anteriormente desconhecidos. O *Deep Learning*, por sua vez, combina avanços no poder computacional e tipos especiais de redes neurais para aprender padrões complicados em grandes quantidades de dados. Já o *Machine Learning*, como vimos, tem como principal área de atuação entender as estruturas dos dados analisados, encaixando distribuições teóricas em dados bem entendidos.

2.4 SIEM - *Security Information and Event Management*

Como já foi dito anteriormente, este trabalho tem por objetivo analisar os logs do processo de *roaming* do eduroam e utilizar o *Machine Learning* para analisar o comportamento dessa troca de

informações, a fim de tomar decisões que auxiliem na manuntenção e melhoria do serviço. Para isso, faremos uso de uma ferrameta que faz parte de um conjunto denominado SIEM (*Security Information and Event Management*), ou Gerência de Informações e Eventos de Segurança, em português.

Trata-se de ferramentas computacionais, processos e procedimentos especializados em coletar, armazenar, processar, monitorar e correlacionar logs de outros sistemas de informações. Sendo assim, serão enviados os logs do *radsecproxy* da federação brasileira, mantida pela RNP, para uma ferramenta que fará toda a análise desses dados recebidos.

os sistemas SIEM surgiram com o objetivo de sanar alguns problemas no gerenciamento de eventos que estavam crescendo paralelamente à evolução da utilização de todos os tipos de aplicações. Alguns pontos que trouxeram à tona a necessidade de sistemas desse tipo foram o crescente número de padrões e normas que obrigam as empresas a reportarem o devido cuidado com relação à segurança da informação e a alta taxa de falsos positivos, provenientes de análises isoladas das soluções, sem a visão de todo o ambiente. Por isso, a solução proposta pelo SIEM traz mudanças significativas como o acesso em tempo real, centralizado e consistente a todos os logs e eventos de segurança, independente do tipo de tecnologia e fabricante. outro ponto muito importante em sistemas SIEM é a utilização de *Machine Learning* para identificação de comportamentos anômalos e criação de alertas e notificações sobre esses comportamento.

Hoje, é possível encontrar diversas ferramentas que utilizam o sistema SIEM. Cada uma delas com especificidades diferentes e prós e contras. A ferramenta SIEM OpenSource mais popular é o AlienVault OSSIM (*Open Source SIEM*), produzida e mantida pela AlienVault. Assim como o OSSIM, tem também o GrayLog, outra ferramenta OpenSource de SIEM. Ambas atendem a diversas demandas de gerenciamento de logs. Uma coisa em comum entre essas duas ferramentas é que as duas utilizam uma ferramenta de indexação chamada Elasticsearch. E, justamente o Elasticsearch, nos trouxe ao conhecimento da ELK Stack, solução SIEM desenvolvida pela empresa Elastic que se baseia na utilização do próprio Elasticsearch, principal produto da Elastic. Essa solução nos pareceu mais conveniente para a execução deste trabalho. Na próxima seção, será explicado melhor sobre a ferramenta e a utilização dela neste projeto, bem como toda a implementação realizada no trabalho.

Capítulo 3

Implementação

Esse capítulo descreve como a arquitetura proposta foi construída, descrevendo as etapas desenvolvimento, os desafios e as limitações de design. A organização do capítulo será primeiramente focada na arquitetura e posteriormente em suas entidades separadamente.

3.1 Ferramenta utilizada

Como vimos na última seção do capítulo 2, diversas empresas tem lançado soluções para análise de dados. Para a realização deste trabalho foi feito um levantamento de qual dessas ferramentas atenderia melhor as necessidades do serviço operado pela RNP. Dentre outros, os principais pontos analisados foram o baixo custo de utilização, a robustez para suportar uma grande massa de dados e a possibilidade de utilização de *Machine Learning*, além de uma documentação única para todas as ferramentas da solução. Assim, a ferramenta identificada como melhor opção foi a solução da empresa *Elastic*, conhecida como ELK Stack.

Trata-se de um conjunto de ferramentas, *Elasticsearch, Kibana e Logstash*, que atuam juntas para uma análise precisa dos dados. Além dessas três principais, a solução também conta com uma outra ferramenta, o Beats, que é módulo instalado no servidor que gera os logs que serão enviados para o ambiente de monitoramento. A figura 3.1 mostra como funciona esse ambiente de monitoramento, onde os logs são enviados pelo módulo do Beats ao Logstash, que faz o parseamento das mensagens e envia para o Elasticsearch, que vai fazer toda aagregação dessas informações, deixando disponíveis para que o Kibana possa mostrar na interface do ambiente.

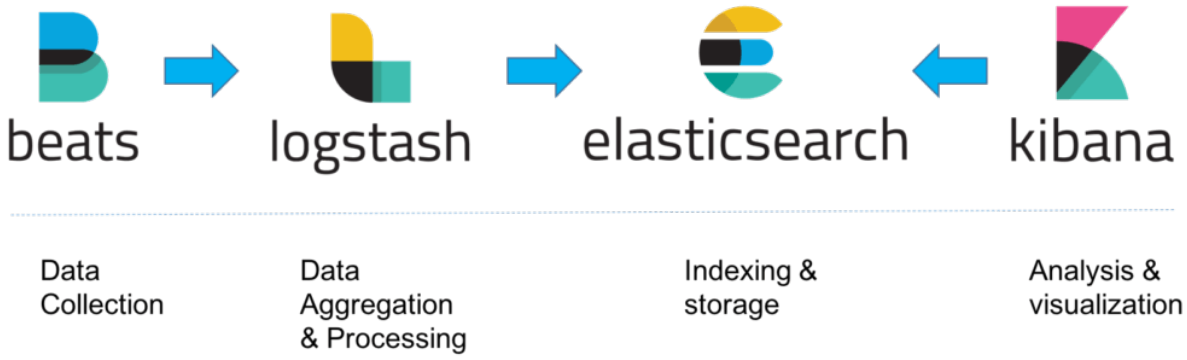


Figura 3.1: Funcionamento da ELK Stack.

3.2 Preparação do Ambiente

Como vimos na seção anterior, a solução utilizada dispõe de três ferramentas principais em sua arquitetura. Para um projeto grande como o que está sendo proposto, a documentação sugere que seja instalado uma ferramenta por máquina virtual. Para a montagem dessa arquitetura, foi utilizado outro serviço da RNP, o compute@RNP, serviço de nuvem para instituições usuárias da RNP.

Foram criadas três máquinas no compute@RNP, todas com a mesma oferta computacional:

- 12GB de memória RAM
- 2TB de disco de armazenamento
- 8 CPU's para processamento

A essas máquinas foram dados os seguintes nomes e endereços IP:

Tabela 3.1: Máquinas criadas e endereços IP

Nome da máquina	Endereço IP
Elasticsearch	200.139.35.11
Logstash	200.139.35.25
Kibana	200.139.35.26

Vimos que é necessária a utilização de uma quarta máquina, a que gera os logs a serem monitorados. No entanto, a máquina que roda esses logs do serviço eduroam já existe, portanto não foi necessário criá-la, apenas configurá-la para enviar os logs.

Os procedimentos realizados para instalação e configuração das ferramentas estão descritos no Anexo A.

3.3 Importação dos dados

Após instalar e configurar as ferramentas nas máquinas criadas, fazendo com que elas se comuniquem entre si para troca de informações, foi iniciado o procedimento para importar os dados do *radsecproxy*. Essa importação foi dividida em duas fases, a importação dos logs armazenados, e a importação dos logs correntes.

3.3.1 Importação de logs armazenados

Esse procedimento foi realizado para que a ferramenta pudesse ter dados suficientes para analisar o comportamento de utilização do serviço, além de nos fornecer também dados estatísticos de uso do serviço. Para que isso pudesse ser realizado, foi disponibilizado pela RNP os logs do ano de 2018. Esses dados ficam armazenados na máquina que roda os logs do *radsecproxy*, separados por arquivos diários. A equipe responsável pela operação dessa máquina compactou esses arquivos e nos disponibilizou para fazermos a análise.

De posse desses dados, foi necessária a criação de uma outra máquina virtual, também utilizando o compute@RNP, com oferta computacional igual às vistas anteriormente. Essa máquina, que recebeu o nome de "Filebeat", nome do módulo usado pela ELK Stack para fazer os envios dos logs. A instalação e configuração desta máquina também está descrita no Anexo A. Essa máquina foi configurada para enviar todos os logs do ano de 2018 para a máquina de entrada do ambiente configurado, no caso a máquina Logstash.

Um ponto interessante na configuração de envio dos logs armazenados é que a ferramenta, por padrão, adiciona os logs com a variável de tempo igual ao tempo de entrada desses logs. Portanto, foi necessária a criação de uma variável denominada "timedate", que armazenasse a data descrita dentro da linha do log. Assim, foi possível receber todos os logs e adicioná-los aos arquivos com o seus respectivos horários.

3.3.2 Importação de logs correntes

A importação de logs atuais, ou seja, em tempo real foi mais simples que a de logs armazenados. Isto porque a ferramenta, por padrão, assume que os logs que estão entrando são atuais, então não é necessária nenhuma configuração adicional. Portanto, bastou configurar a máquina *radsecproxy* para enviar os logs em tempo real.

Para que isso fosse realizado, no entanto, foi preciso passar por um processo um pouco mais burocrático dentro da RNP. Tendo em vista que essa é uma máquina rodando no ambiente de produção e responsável por um serviço de porte grande, foi necessário mostrar toda a configuração no ambiente de homologação primeiramente e, depois disso, abrir uma solicitação de mudança para que a coordenação da equipe responsável autorizasse a instalação do módulo do Filebeat dentro dessa máquina. Depois que foi autorizado, foi passado como deveria ser feita a instalação e configuração do Filebeat para enviar os logs para o ambiente de monitoramento e o procedimento foi realizado pela equipe responsável.

Uma vez que tínhamos todos os logs necessários para fazer a análise, foi iniciado o trabalho de análise dos dados, começando pela análise estatística da utilização do eduroam e, posteriormente, fazendo a análise comportamental, utilizando o módulo de *Machine Learning* disponibilizado pela Elastic.

3.3.3 Discover

Para visualizarmos as entradas de todos esses dados na interface do Kibana, devemos acessar a aba Discover. A Figura 3.2 mostra como fica essa ppágina, visualizando todas as entradas desde o início de 2018. Explicando melhor a página, temos alguns elementos que serão importantes. À esquerda tem o menu da ferramenta, que mostra todas as opções de ações que a solução nos oferece, como a página de Dashboard e a de *Machine Learning*, duas páginas que usaremos neste trabalho. O menu superior nos mostra as ações disponíveis para a página que estamos. A mais importante configuração nesse menu é a opção de escolher a data para visualizar os dados. Como foi dito, utilizaremos, neste trabalho, o ano de 2018. O gráfico mostrado na página evidencializa a contagem de dados com o passar do tempo. O conteúdo abaixo do gráfico, juntamente com o menu à esquerda do mesmo, são conteúdos das mensagens de log. O menu à esquerda mostra as variáveis capturadas dentro da mensagem, nos dando a opção de selecioná-las para a melhor visualização abaixo do gráfico, onde são mostradas as variáves escolhidas.

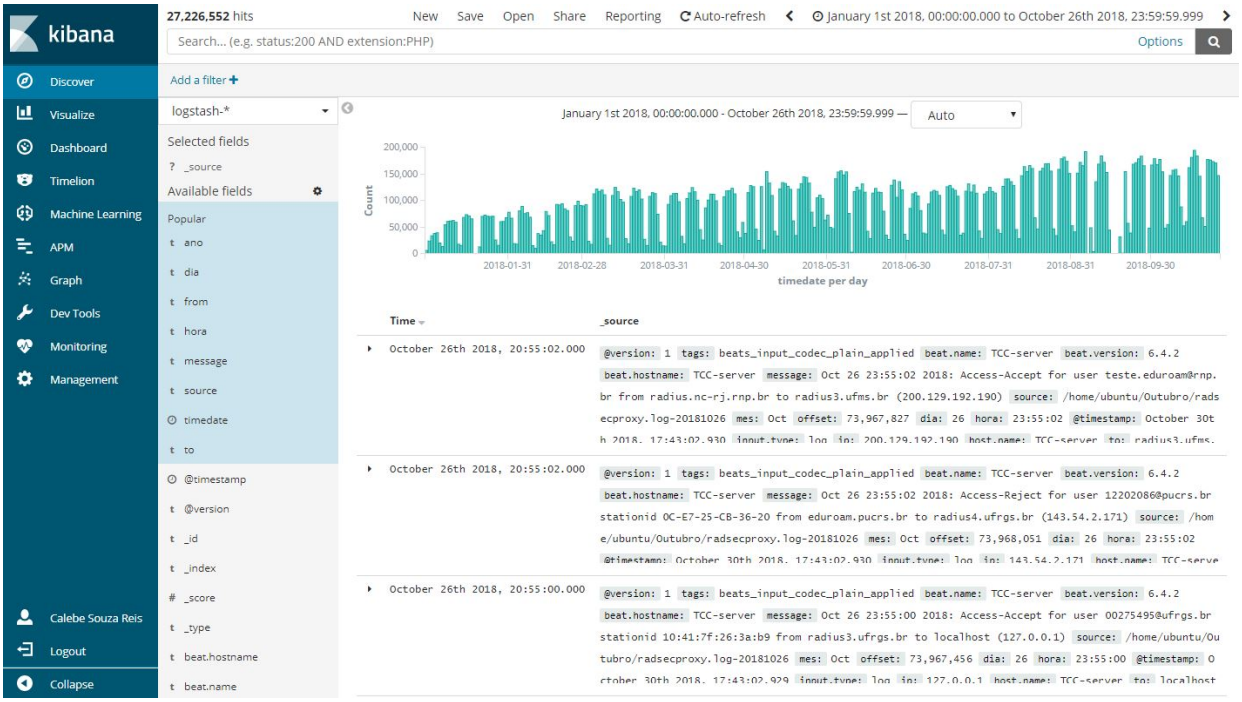


Figura 3.2: Página de visualização das entradas.

3.4 Criação de Visualizações

Com os dados importados para a ferramenta, criaremos visualizações, em vários formatos, para melhor analisarmos os dados coletados. A solução implementada nos fornece diversas opções, porém utilizaremos apenas algumas delas. A Figura 3.3 nos mostra todas essas opções disponíveis.

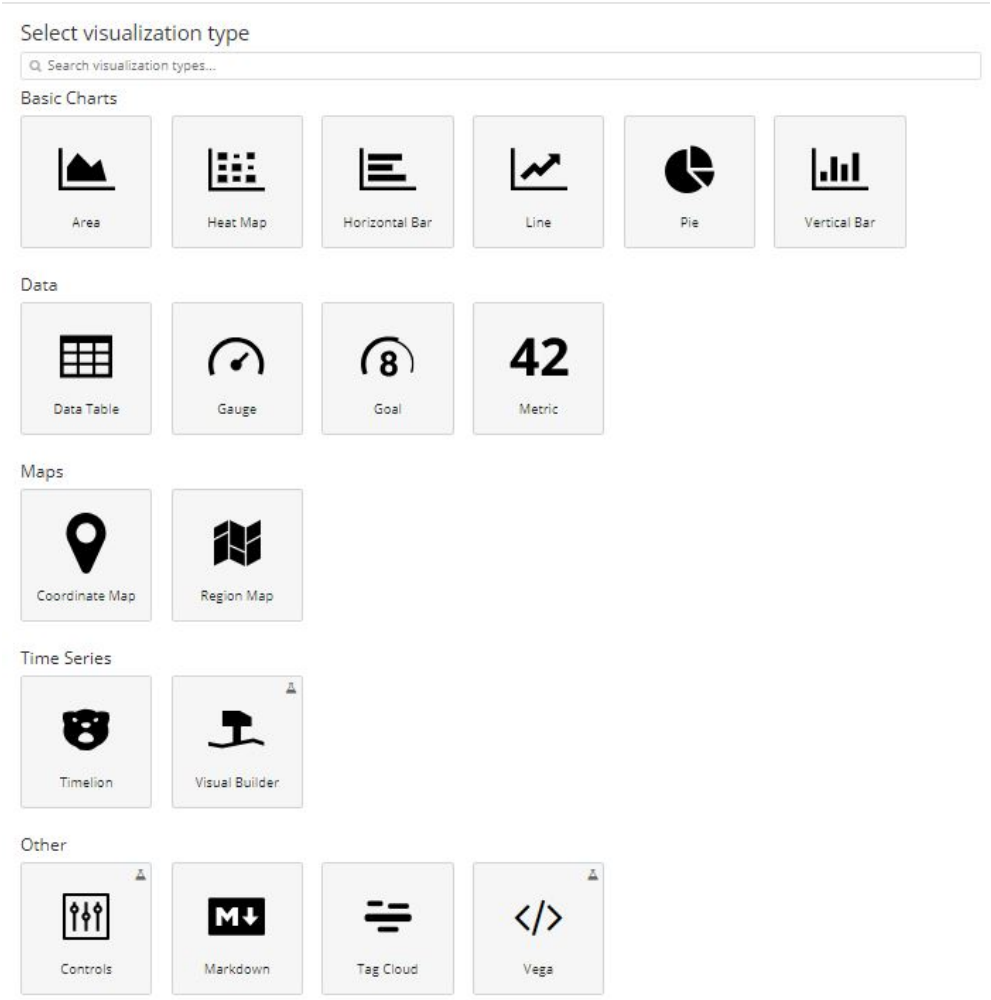


Figura 3.3: Tipos de visualização disponíveis.

Para a elaboração desta prova de conceito, foram selecionados quatro tipos de visualização, que serão melhor apresentados a seguir.

Para todas as opções, quando selecionamos a visualização desejada, abre-se uma página, mostrada na Figura 3.4, para selecionarmos de qual entrada ou de qual busca salva na aba Discover deseja-se analisar os dados. No caso deste trabalho, a seleção será sempre pelo index "logstash-*", que seleciona todos os indexes provenientes da máquina Logstash. Portanto, será omitido esse passo nas seções seguintes.

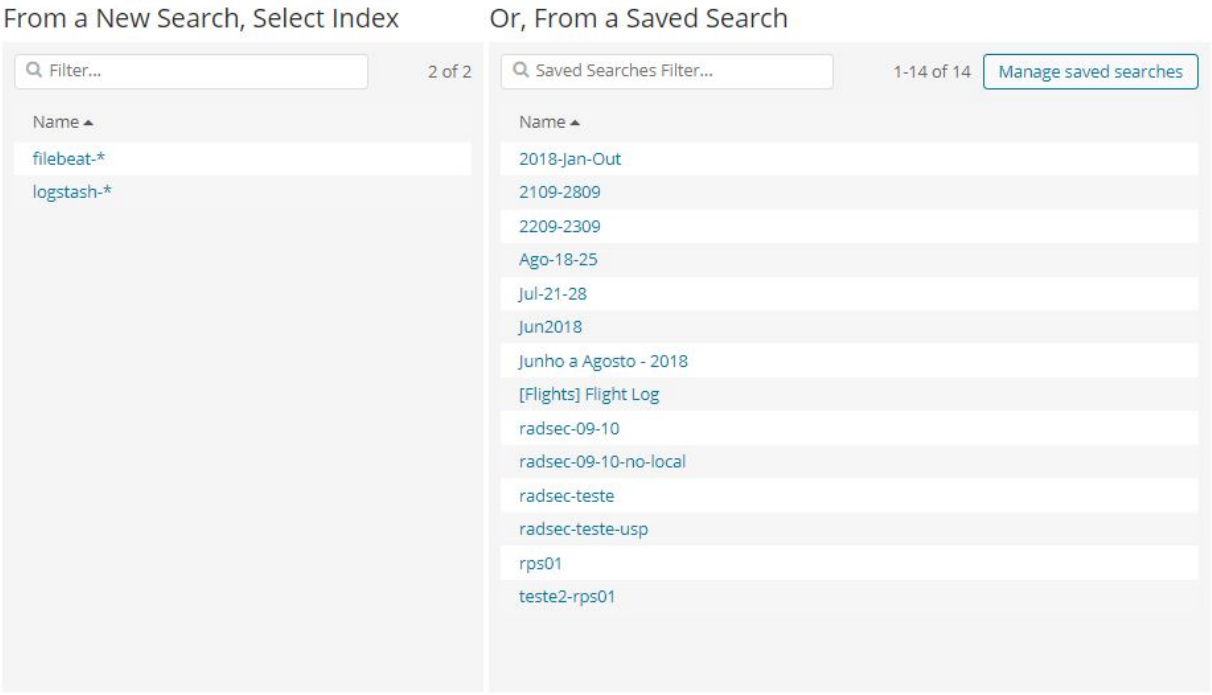


Figura 3.4: Página para selecionar o index desejado.

3.4.1 Métricas

Com a visualização de métricas, os dados que serão coletados são as quantidades de requisições realizadas no período selecionado, além do quantitativo de aceitações ou rejeições dessas requisições. A Figura 3.5 mostra quais opções de configuração é possível utilizar nesse tipo de visualização.

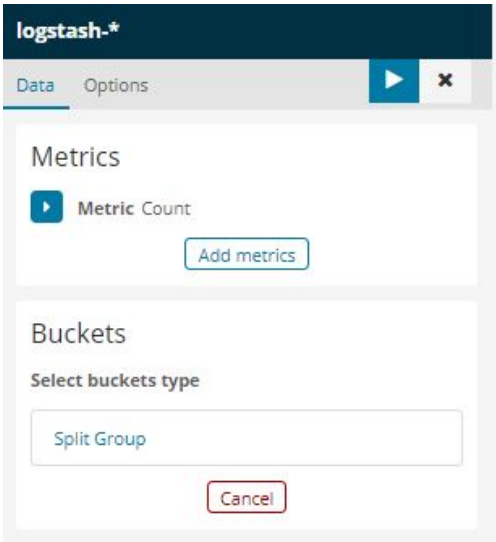


Figura 3.5: Configurações disponíveis para a visualização de métricas.

Para a primeira visualização desejada, da contagem geral de requisições, não é necessário fazer

nenhuma alteração nas configurações, pois, por padrão, essa visualização conta os documentos na entrada. Já para visualizar a quantidade de requisições aceitas e rejeitadas é preciso aplicar um filtro, buscando pelas palavras "Accept"e "Reject". Para fazer isso, clicamos na opção Split Group, escolhemos a opção "filters"e digitamos o valor desejado. A Figura 3.6 exemplifica essa configuração explicada.

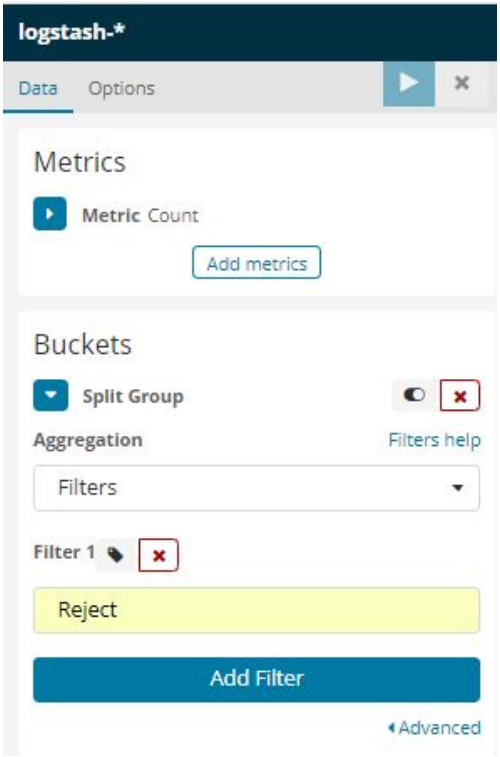
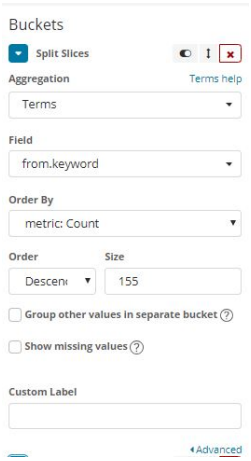


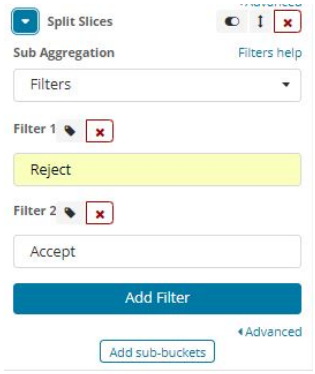
Figura 3.6: Configuração utilizando filtros.

3.4.2 Gráficos de Pizza

Com os gráficos do tipo "pizza", nomeado na ferramenta de "Pie", analisaremos quais são as instituições mais participantes no proceso de *roaming* do eduroam. Para isso, iniciaremos da mesma forma anterior, escolhendo a visualização desejada e o index "logstash-*". A Figura 3.7 mostra as opções de configuração possíveis. para exemplificar, vamos montar a visualização com as instituições que mais receberam requisições em 2018. Para isso, na seção Buckets, selecionamos a opção *Split Slices*. No campo *Aggregation*, escolheremos a opção "terms"e o campo "from.keyword". Por último, vamos separar ainda num anel externo, dividindo as requisições entre aceitas e rejeitadas, utilizando a opção *Add sub-buckets*, depois, mais uma vez, a opção *Split Slices*. Porém, no campo *Aggregation*, dessa vez escolheremos *filters*, e selecionaremos dois filtros: "Accept"e "Reject". Essa configuração do gráfico de é mostrada na Figura 3.8. O resultado desse gráfico, assim como os outros, será mostrado no Capítulo 4, quando analisaremos os resultados.



(a) Separação por termos



(b) Separação por filtro

Figura 3.8: Configuração para gráfico de pizza

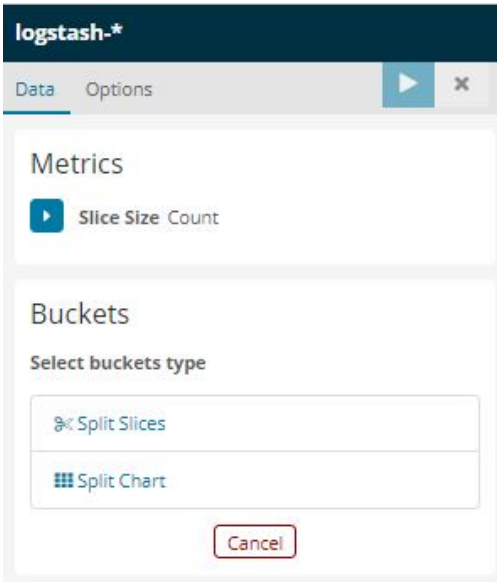
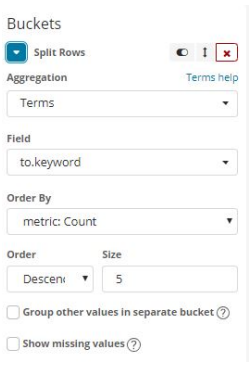


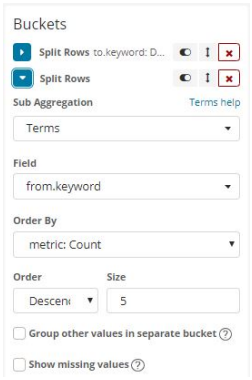
Figura 3.7: Configuração do gráfico de pizza.

3.4.3 Tabelas

Para montar tabelas como visualização, a opção a ser escolhida é "Data Table", e a entrada de dados é, mais uma vez, o index "logstash-*". Para mostrar a configuração das tabelas, usaremos o exemplo das maiores conexões entre instituições. A Figura 3.9 mostra as opções disponíveis para configuração de tabelas. Precisaremos escolher o campo *Split Rows*, que são as colunas. Criaremos então duas colunas, além da coluna padrão de contagem de documentos. Em uma coluna, selecionaremos o campo "to.keyword"e, na outra coluna, o campo "from.keyword". Essa configuração é mostrada na Figura 3.10.



(a) Primeira separação por termos



(b) Segunda separação por termos

Figura 3.10: Configuração para tabela.

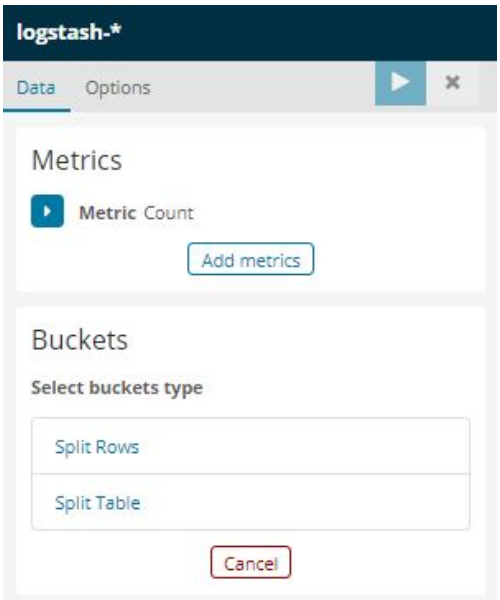


Figura 3.9: Opções de configuração para Tabelas.

3.4.4 Linha do Tempo

A visualização por linha do tempo é um pouco diferente das outras mostradas acima. A ELK Stack tem uma aba só para esse tipo de visualização, o *Timelion*. Para montarmos um gráfico no *Timelion*, é necessário fazer uso de uma expressão regular, específica para essa ferramenta. A Figura 3.11 mostra o layout da página de configuração. No menu superior da página tem uma opção de ajuda, mostrando todas as opções de sintaxe disponíveis. Como exemplo, montaremos um gráficos com duas linhas do tempo, tomando a USP como instituição a ser analisada. A primeira delas mostra a contagem geral de requisições feitas pela USP com o passar do tempo, enquanto a segunda linha mostra quantas dessas requisições foram rejeitadas.

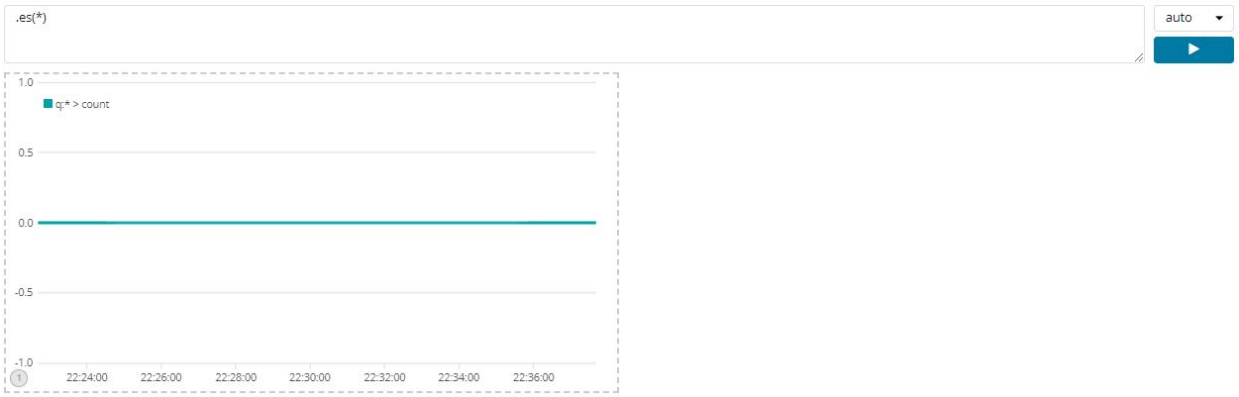


Figura 3.11: Página de configuração *Timelion*

A expressão abaixo foi a utilizada para obter o resultado descrito acima.

```
.es(index=logstash-*, timefield=timedate,q='to:eduroam01.semfio.usp.br
OR to:eduroam02.semfio.usp.br',q='to:eduroam01.semfio.usp.br OR
to:eduroam02.semfio.usp.br AND Reject')
```

3.5 Análise Comportamental

Para configuração da análise comportamental dos dados de entrada, utilizaremos a página de *Machine Learning* disponível no menu lateral do Kibana. Realizaremos duas análises, uma com o foco na situação das instituições realizando requisições, e outra com o foco nas instituições recebendo requisições. Para exemplificar a configuração, utilizaremos a primeira análise. A Figura 3.12 mostra a página inicial do *Machine Learning*, onde lista as análises já realizadas e nos dá a opção de realizarmos novas. Para realizarmos uma nova análise, clicamos no botão "Create New Job", selecionamos o index "logstash-*"e, na próxima página, mostrada da Figura 3.13, que nos dá a opção de vários tipos de análise, selecionamos a opção "Multi metric", que detecta anomalias em diferentes métricas a partir da separação dos dados do campo definido. A Figura 3.14 mostra a página de configuração da análise. Para a análise escolhida, os campos selecionados serão o "event rate", para levar em consideração o tempo, e o "to.keyword", para selecionar as instituições que realizaram requisições. Para separar os dados de entrada na variável "to", escolhemos a opção "to.keyword"no campo "Split Data". Depois, adicionamos um nome e descrição para a análise e a inicializamos. As configurações selecionadas podem ser vistas na Figura 3.15.

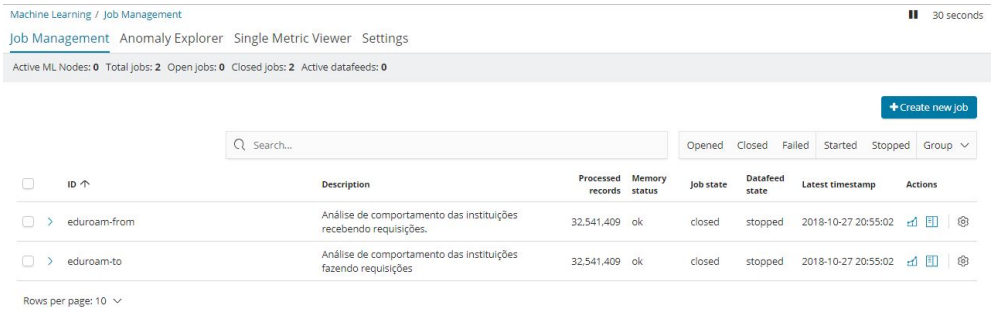


Figura 3.12: Página inicial do *Machine Learning*.

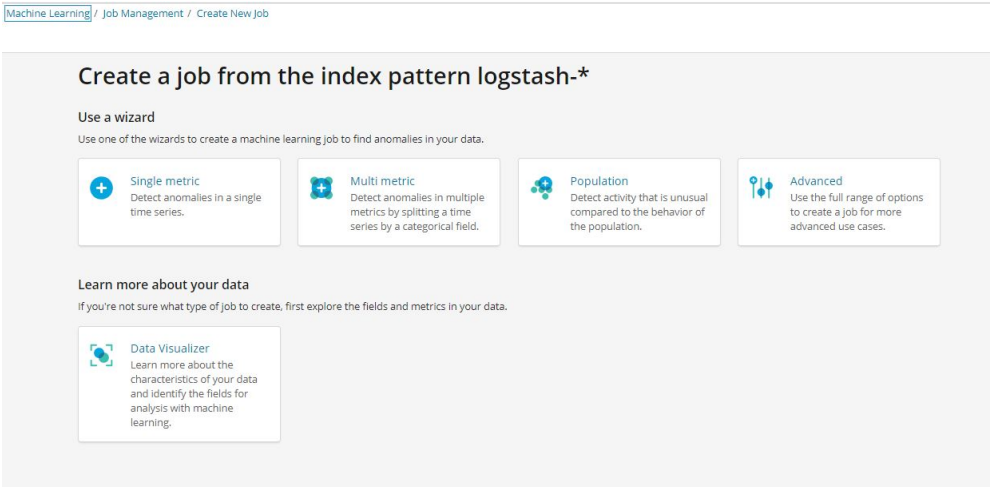


Figura 3.13: Opções de análise disponíveis.

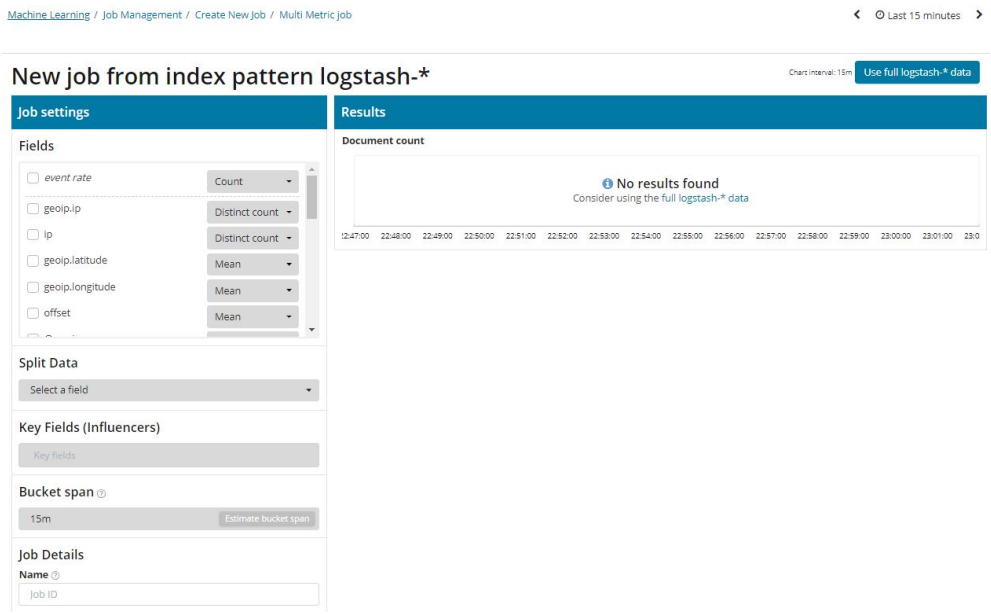


Figura 3.14: Opções de configuração de uma análise no *Machine Learning*.

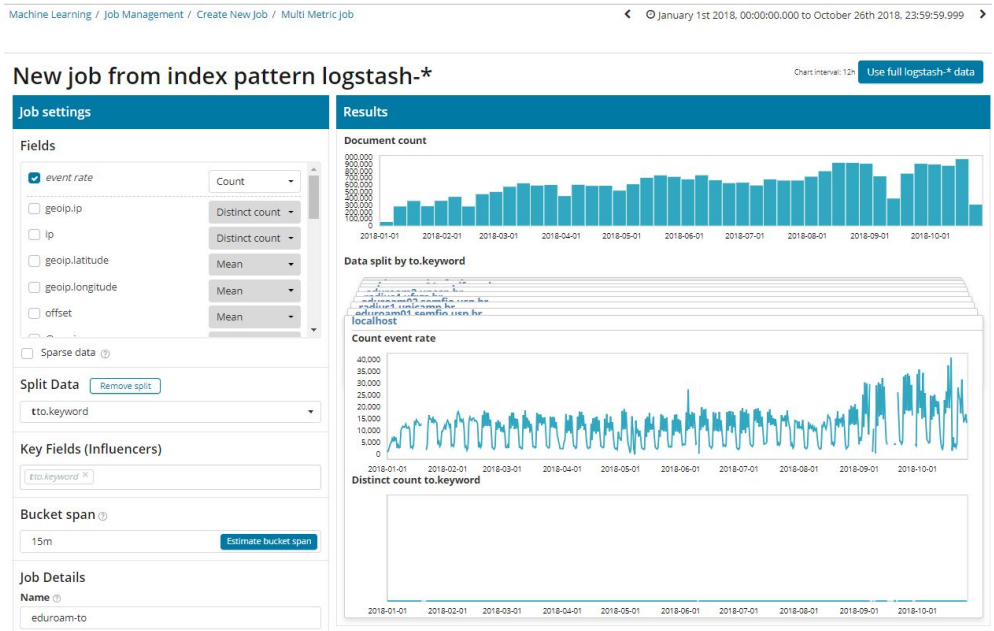


Figura 3.15: Configurações escolhidas para a análise.

3.6 Criação de alertas

Para a criação de alertas, levando em consideração as anomalias detectadas a partir da análise descrita na seção anterior, utilizaremos uma opção chamada *Watcher*. Para acessá-la, devemos acessar a aba *Management*, no menu lateral e, no campo Elasticsearch, clicar na opção *Watcher*. A Figura 3.16 mostra a página inicial da opção *Watcher*.

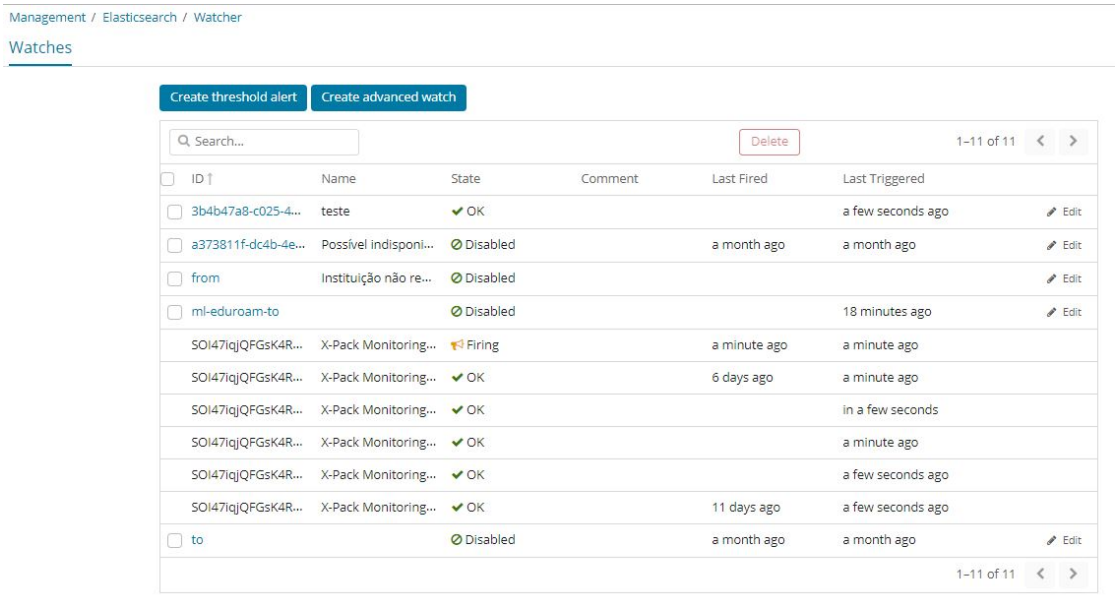


Figura 3.16: Página de visualização de *Watchers*.

Configuraremos três opções de alerta para o monitoramento do eduroam. O primeiro alerta é criado escolhendo a opção *Create Treshold alert* e pode ser configurado graficamente, como veremos a seguir. Os dois outros alertas são criados a partir de um código JSON e, por isso, é na opção *Create Advanced Watch*. Os códigos escritos para esses alertas estão apresentados no Anexo B.

O primeiro alerta a ser criado notificará o administrador do serviço eduroam quando o *radsec-proxy* estiver a mais de 10 minutos sem receber nenhuma requisição. Isso indicará uma possível queda do serviço. Para fazermos tal configuração, clicamos na opção *Create Treshold alert*. Na página mostrada na Figura 3.17, configuramos quando o alerta será criado. Já na página mostrada na Figura 3.18, configuramos a ação realizada quando a análise retornar um resultado anômalo. No caso, configuraremos uma mensagem simples de e-mail a ser enviada para o administrador do serviço.

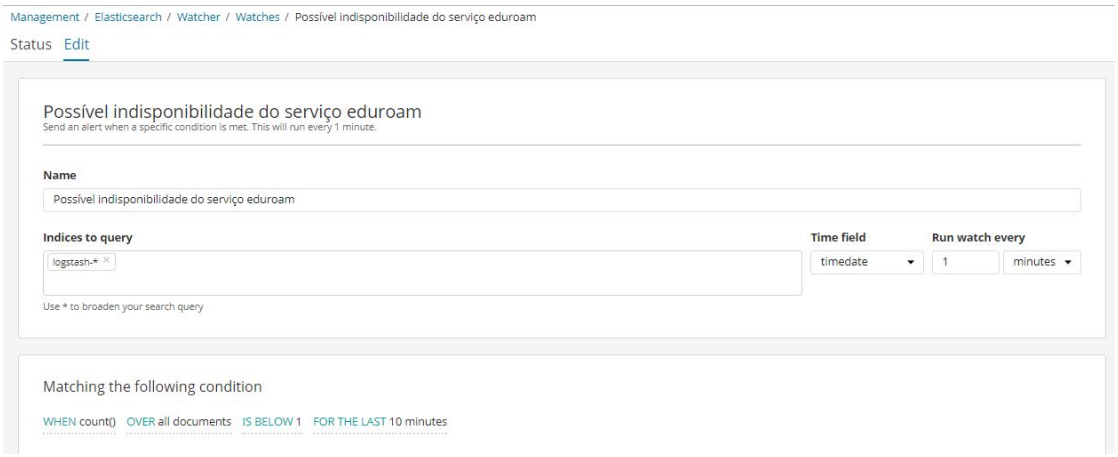


Figura 3.17: Configuração do *Watcher*, parte 1.

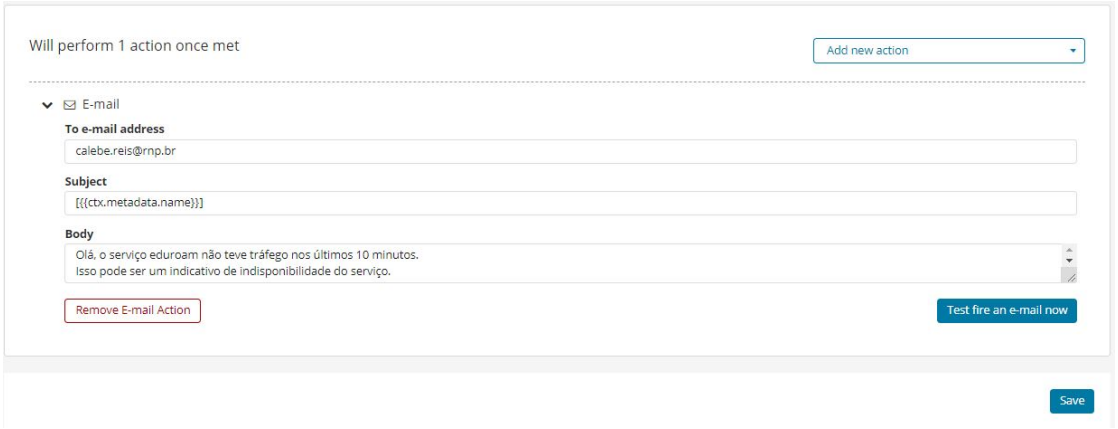


Figura 3.18: Configuração do *Watcher*, parte 2.

Capítulo 4

Análise e Resultados

Como vimos no capítulo anterior, foram implementadas três funcionalidades importantes para o monitoramento do processo de *roaming* do eduroam, a criação de dashboards e visualizações dos dados coletados, a análise comportamental do serviço eduroam e das instituições participantes e a criação de alertas a partir dessa análise comportamental. Portanto, este capítulo é dedicado à apresentação dos resultados obtidos com a implementação dessas funcionalidades que compõem a prova de conceito realizada. Para efeito de análise, foi selecionado o período de Janeiro a Outubro de 2018. Todas as informações que serão apresentadas a seguir dizem respeito a esse período.

4.1 Visualização dos dados

Nesta primeira seção, serão analisadas e apresentadas as visualizações criadas para a elaboração deste trabalho. No capítulo 3 foi mostrado que a ELK Stack nos dá muitas opções de visualizações. Porém, para a realização da prova de conceito, foram selecionadas apenas algumas dessas opções. Para cada uma das visualizações escolhidas, será apresentado um, ou mais, exemplo de como foi utilizada e uma análise sobre o dado apresentado.

4.1.1 Métricas

A visualização de métricas pode ser utilizada para apresentar dados numéricos sobre determinada ação. No caso do monitoramento que está sendo realizado, essa visualização apresenta, como pode ser visto na Figura 4.1, a quantidade de requisições realizadas no período determinado, além de quantas dessas requisições foram aceitas e quantas foram rejeitadas. Esse dado é genérico, não diz respeito a uma instituição em especifíco, mas ao serviço como um todo. Portanto, o que podemos aferir desses dados é que o serviço eduroam tem uma relação entre requisições aceitas e rejeitadas que pode ser considerada boa, superior a 70%. As requisições rejeitadas podem ser provenientes de diversos motivos, como erro na digitação de login ou senha, por exemplo. Com a análise individual, que será mostrada mais a frente, poderemos verificar também se alguma instituição está com essa relação mais preocupante e, assim, agir de alguma forma para tentar

melhorá-la.

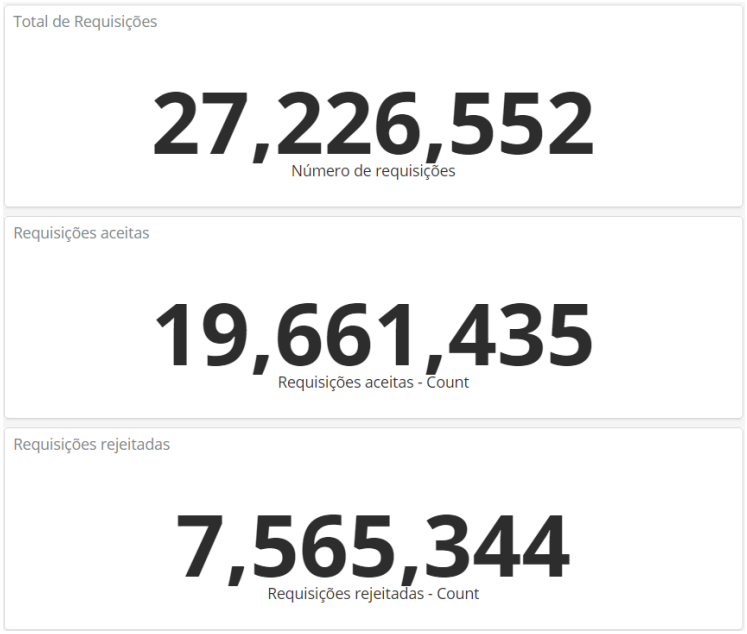


Figura 4.1: Quantidade de requisições. Requisições aceitas e Requisições rejeitadas.

4.1.2 Gráficos de Pizza

Os gráficos de "pizza", como são conhecidos, são exemplos de visualizações muito utilizadas em análise de todos os tipos de dados. Neste trabalho, serão medidas, a partir desses gráficos, as porcentagens das instituições no processo de *roaming*, mostrando quais instituições mais fizeram e quais mais receberam requisições, além de mostrar a relação entre requisições aceitas e rejeitadas por instituição.

As Figuras 4.2 e 4.3 mostram esses dados a partir da ótica da relação entre requisições aceitas e rejeitadas, evidencializando, também, quais são as instituições que tem mais requisicões aceitas e rejeitadas, tanto quando estão fazendo as requisições (Figura 4.2), quanto quando estão recebendo requisições (Figura 4.3).

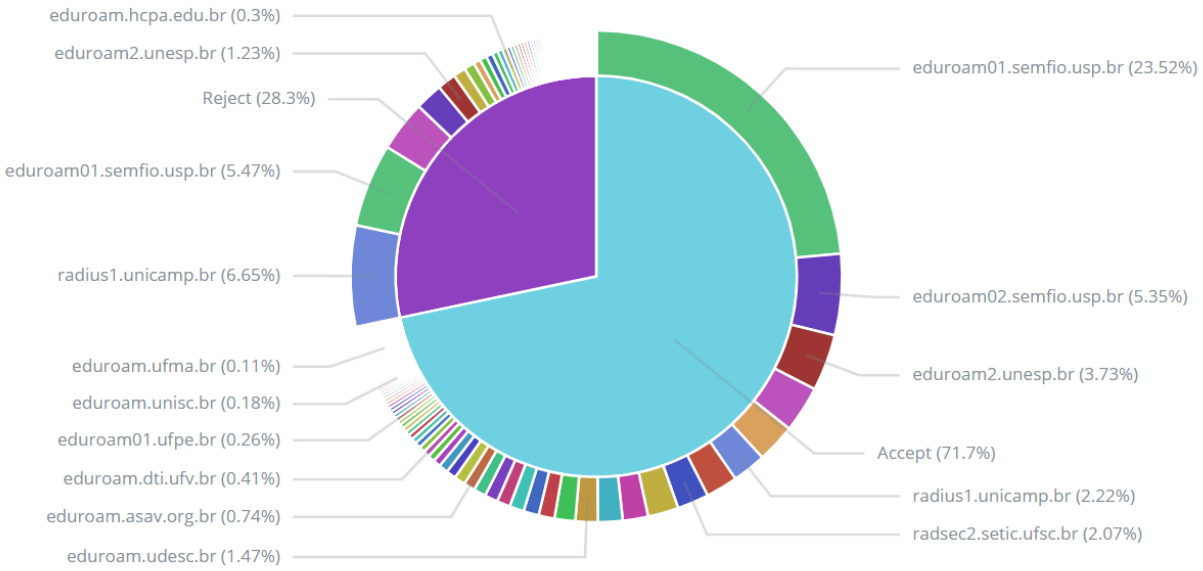


Figura 4.2: Relação da quantidade de requisições aceitas e rejetadas quando as instituições estão recebendo visitantes.

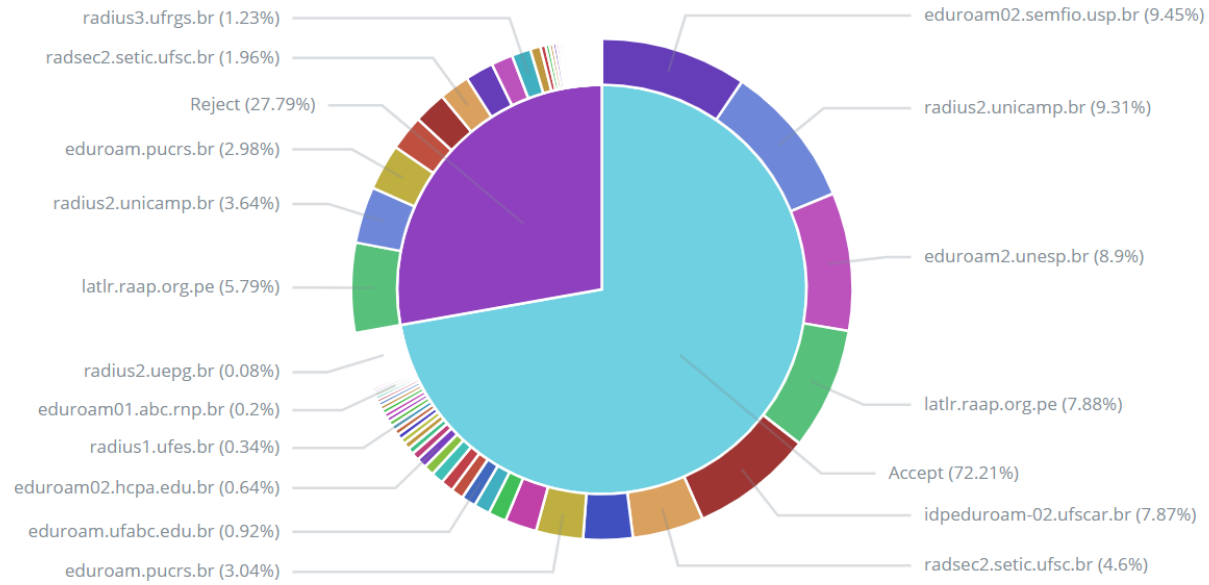


Figura 4.3: Relação da quantidade de requisições aceitas e rejetadas quando as instituições estão enviando visitantes.

As Figuras 4.4 e 4.5 mostram os mesmos dados das imagens anteriores, porém com um ponto de vista diferente. Podemos observar que o gráfico interno agora mostra as instituições e o externo, as quantidades de *Accepts* e *Rejects*.

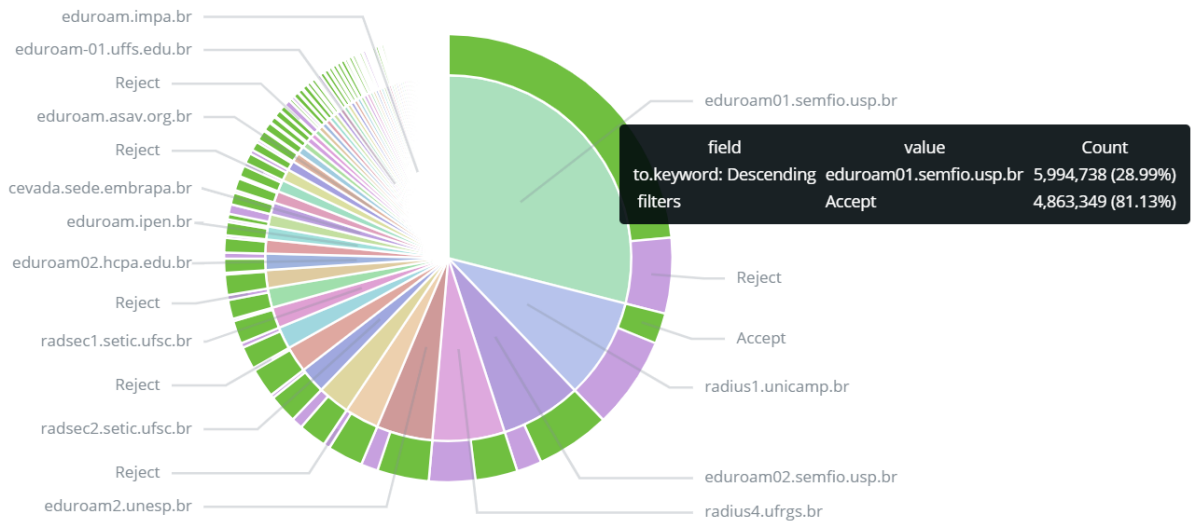


Figura 4.4: Instituições que mais fizeram requisições e os valores relacionados.

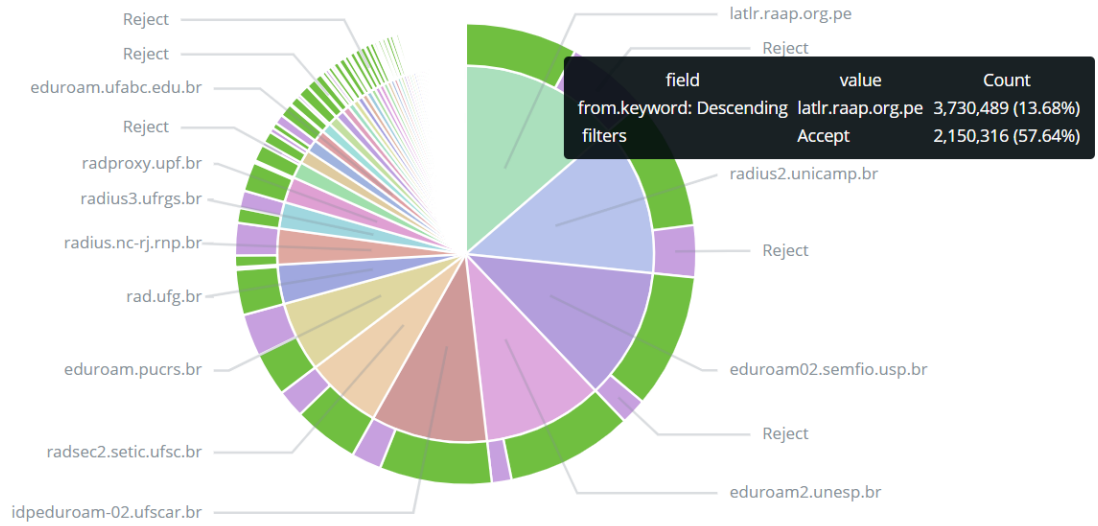


Figura 4.5: Instituições que mais receberam requisições e os valores relacionados.

A partir desses gráficos é possível verificar quais são as instituições mais ativas no processo de *roaming*. Entre elas, vale destacar a Universidade de São Paulo (USP), que detém uma porcentagem significante das requisições realizadas. Há ainda um aumento nesse domínio da USP quando observamos que a instituição tem dois servidores, eduroam01 e eduroam02, ativos nesse processo. Outro dado importante que podemos ver é que a Universidade de Campinas (UNICAMP - SP) possui uma taxa de requisições rejeitadas muito maior que aceitas. Esse é um dado muito importante para tentarmos identificar possíveis causas desse problema para auxiliar a instituição na resolução do mesmo.

Dentre as instituições que mais recebem requisições, podemos observar no gráfico um endereço

não brasileiro com a maior porcentagem. Trata-se do servidor da Confederação Latino Americana (LATLR), pois o *radsecproxy* da federação do Brasil, quando identifica qualquer domínio que não pertence a sua alçada, encaminha para a LATLR. Portanto, o que podemos afirmar é que essas requisições são internacionais, mas não podemos afirmar de quais instituições especificamente. Entre as insituições brasileiras, a que mais recebe requisições, ou seja, mais envia usuários a outras instituições, é a UNICAMP.

4.1.3 Tabelas

Outra visualização escolhida para ser utilizada neste trabalho foi a de tabelas. No caso específico do monitoramento do eduroam, foram construídas tabelas para fazer diferentes listagens. Os gráficos de pizza, vistos na seção anterior, também poderiam ser visualizados na forma de tabelas. As tabelas vistas nas Figuras 4.6 e 4.7 mostram as instituições menos ativas no ano de 2018, tanto fazendo quanto recebendo requisições.

Menos fizeram requisições

to.keyword: Ascending	Count
temp.info.ufrn.br	1
eduroam.ufsj.edu.br	2
radius3.ufrgs.br	2
idp-eduroam.uerj.br	3
frankfurt.ifrn.edu.br	5
eduroam.cnpq.br	6
eduroam.uftm.edu.br	6
eduroamjmv.ufop.br	6
eduroam.ifto.edu.br	9
eduroam.ifsp.edu.br	10

Export: Raw Formatted

Figura 4.6: Instituições que menos fizeram requisições.

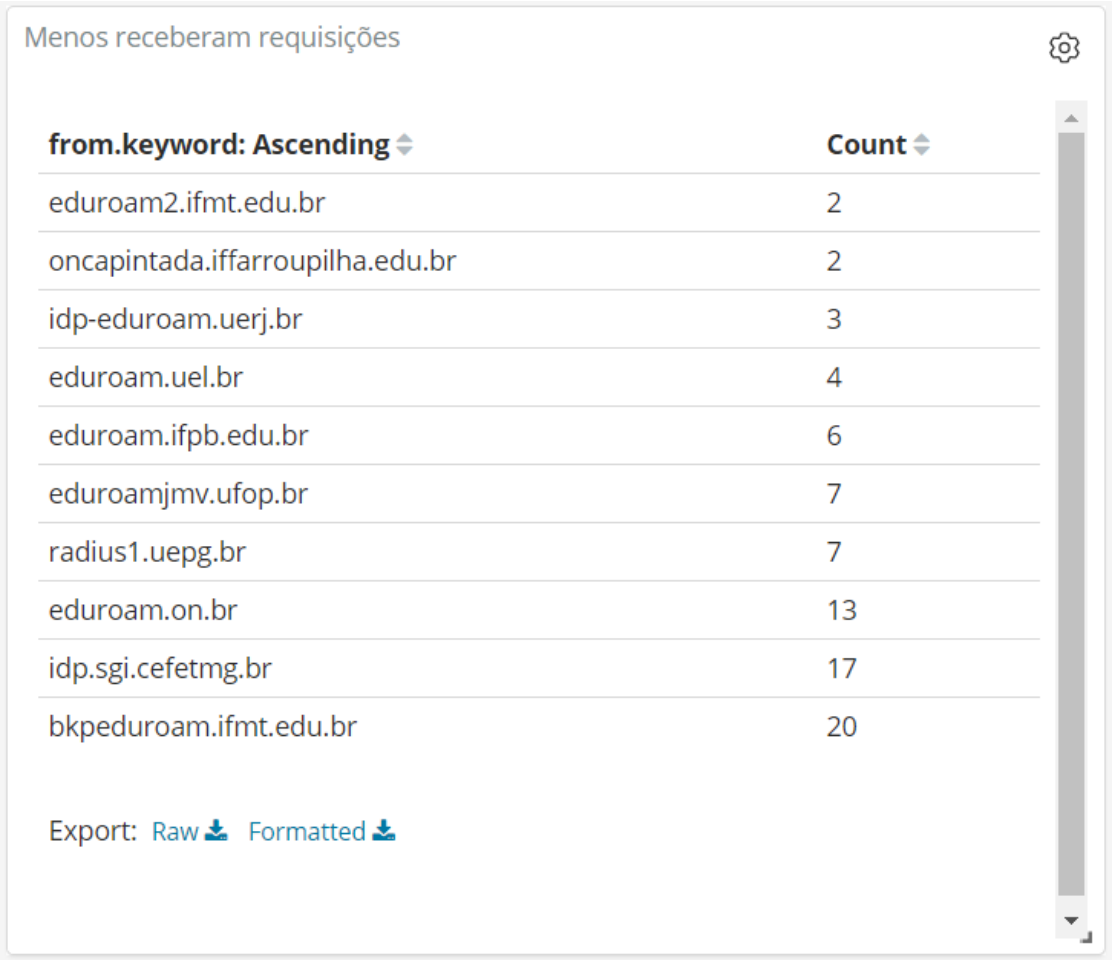
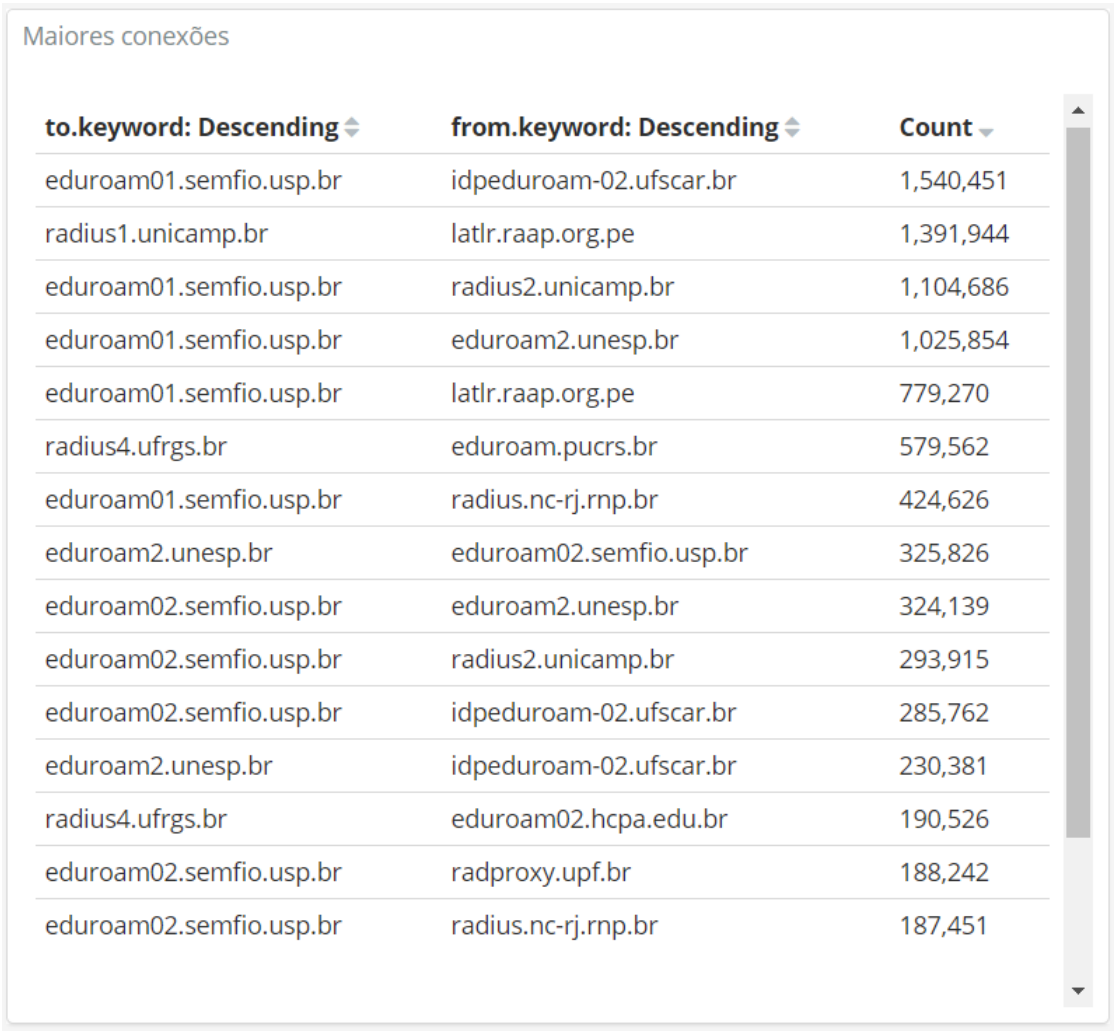


Figura 4.7: Instituições que menos receberam requisições.

A Figura 4.8, por outro lado, traz um dado muito interessante para a nossa análise, mostrando quais são as maiores conexões entre instituições. Podemos observar que a maior conexão entre instituições é entre a USP, que, como vimos, é a instituição que mais faz requisições, e a UFSCar (Universidade Federal de São Carlos). Analisando este dado, podemos aferir que isso se deve ao fato de que essas universidades são muito próximas e, talves por isso, muito usuários da UFSCar utilizem o eduroam na USP. Também é possível observar que a UNICAMP recebe muitos usuários de instituições internacionais, fazendo com que a conexão entre ela e a LATLR seja responsável por muitas requisições.



Maiores conexões

to.keyword: Descending	from.keyword: Descending	Count
eduroam01.semfio.usp.br	idpeduroam-02.ufscar.br	1,540,451
radius1.unicamp.br	latlr.raap.org.pe	1,391,944
eduroam01.semfio.usp.br	radius2.unicamp.br	1,104,686
eduroam01.semfio.usp.br	eduroam2.unesp.br	1,025,854
eduroam01.semfio.usp.br	latlr.raap.org.pe	779,270
radius4.ufrgs.br	eduroam.pucrs.br	579,562
eduroam01.semfio.usp.br	radius.nc-rj.rnp.br	424,626
eduroam2.unesp.br	eduroam02.semfio.usp.br	325,826
eduroam02.semfio.usp.br	eduroam2.unesp.br	324,139
eduroam02.semfio.usp.br	radius2.unicamp.br	293,915
eduroam02.semfio.usp.br	idpeduroam-02.ufscar.br	285,762
eduroam2.unesp.br	idpeduroam-02.ufscar.br	230,381
radius4.ufrgs.br	eduroam02.hcpa.edu.br	190,526
eduroam02.semfio.usp.br	radproxy.upf.br	188,242
eduroam02.semfio.usp.br	radius.nc-rj.rnp.br	187,451

Figura 4.8: Maiores conexões entre instituições.

4.1.4 Linha do tempo

O último tipo de visualização escolhido para compor este trabalho é a partir de linha do tempo. Na ELK Stack, se dá o nome de *Timelion* para este tipo de visualização. Com esse gráfico, conseguimos observar o comportamento das instituições com o passar do tempo. Para exemplificar, as Figuras 4.9 e 4.10 mostram o comportamento da USP durante o ano de 2018. Como trata-se da universidad que mais utiliza o eduroam no Brasil, podemos observar que a quantidade de requisições é bastante elevada. Nos dois gráficos, a linha verde signifca a quantidade de requisições e a vermelha, a quantidade de rejeições dessas requisições. É possível afirmar que a relação entre aceitações e rejeições está em um nível considerado normal. Também podemos ver que a USP utiliza muito mais o eduroam para fazer requisições do que para recebê-las, isto significa que muitos usuários de outras instituições visitam a USP, mas a quantidade de usuários de lá que visitam outras instituições é bem menor.

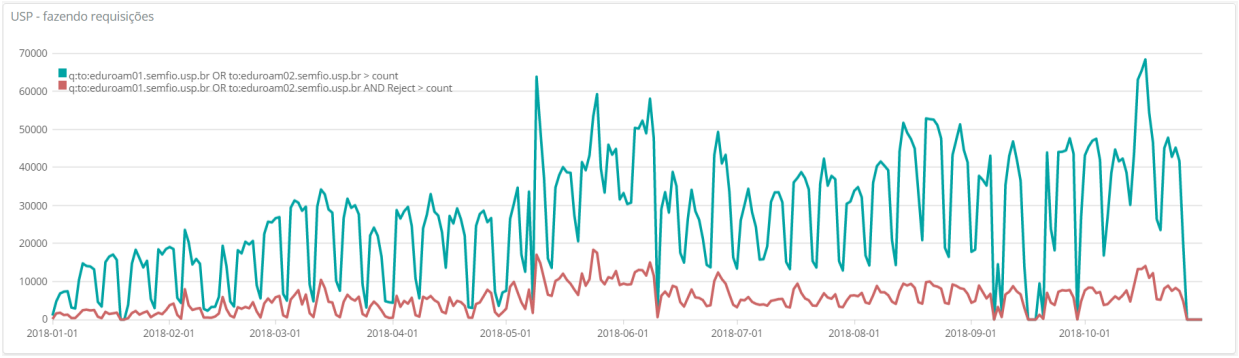


Figura 4.9: USP - Requisições feitas e requisições rejeitadas.

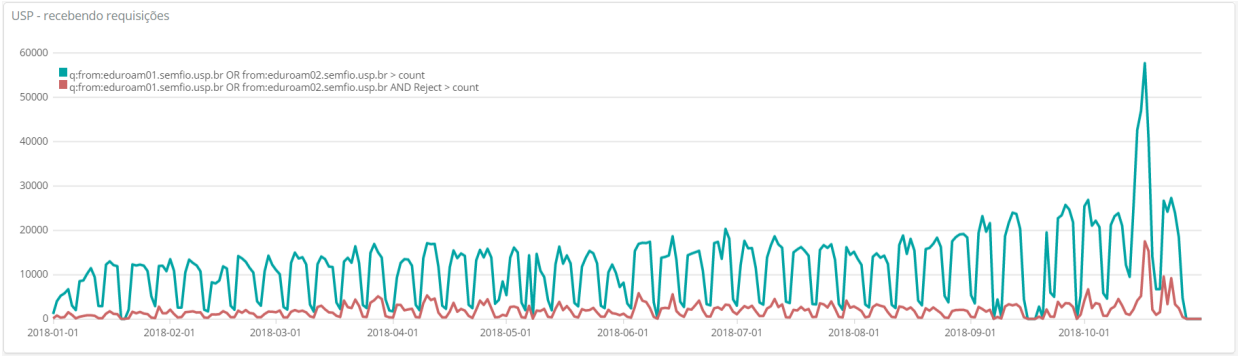
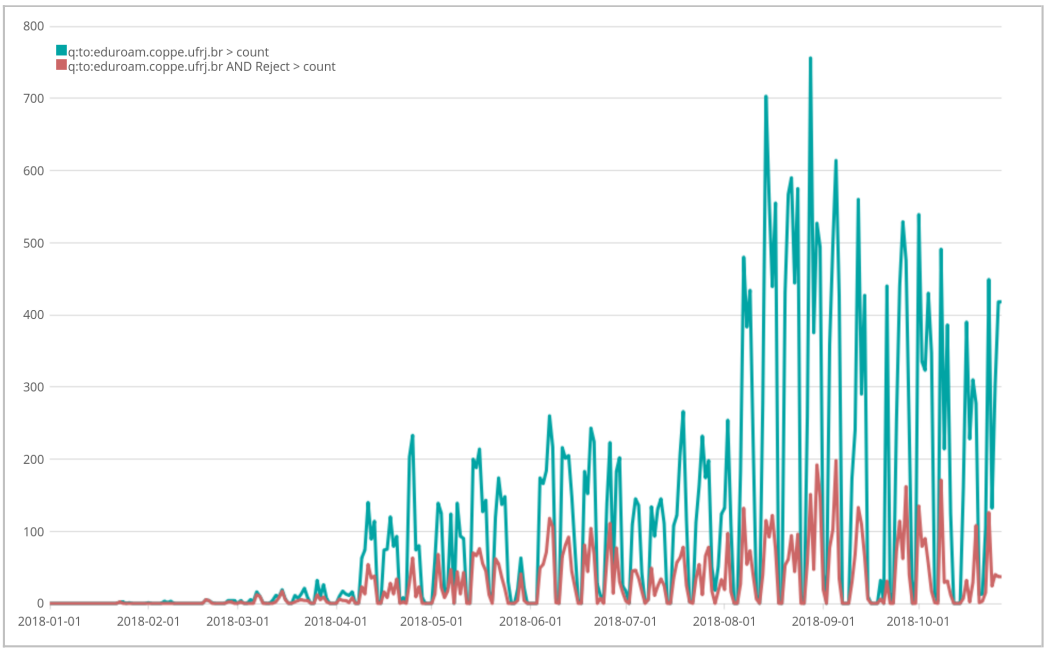


Figura 4.10: USP - Requisições recebidas e requisições rejeitadas.

4.1.5 Geração de relatórios

A ferramenta ELK Stack também nos permite criarmos relatórios a partir de visualições ou dashboards. A Figura 4.11 traz um modelo de como fica um relatório gerado diretamente da ferramenta, a partir da função *Reporting*, já em modelo PDF. Para a geração deste relatório foi utilizada a visualização do tipo *Timelion*, como vimos na seção anterior, para mostrar a utilização da instituição COPPE-UFRJ durante o ano de 2018, bem como a quantidade de rejeições das requisições realizadas.

Coppe UFRJ - Requisições realizadas



Coppe UFRJ - Requisições recebidas

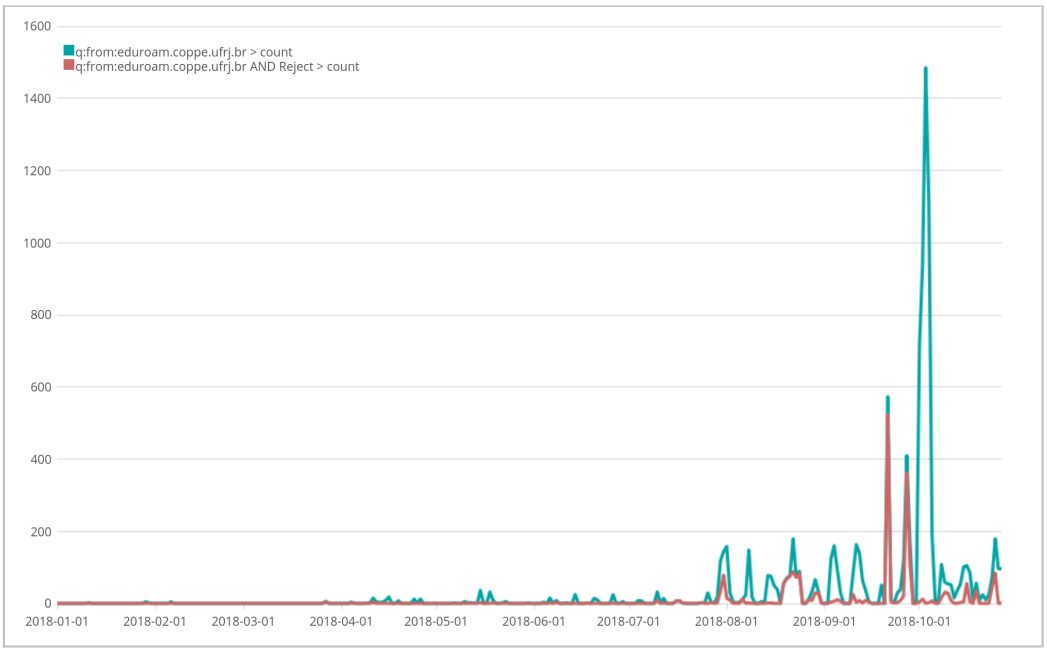


Figura 4.11: Relatório gerado sobre a instituição COPPE-UFRJ.

4.2 Análise Comportamental

Esta seção mostrará os resultados das análises comportamental realizadas. As análises foram realizadas em cima de todos os arquivos de logs enviados e, por isso, é possível ter uma análise geral, do serviço como um todo, e outra individualizada, analisando o comportamento de cada instituição. Foram realizadas duas análises, uma para as instituições recebendo requisições e outra para a utilização do eduroam pelas intituições quando estão fazendo requisições.

4.2.1 Visão Geral

A análise realizada na ELK Stack nos traz resultados que podem ser visualizados de duas formas. A primeira forma é chamada de *Anomaly Explorer*, ou Explorador de Anomalias. Este modo traz quais são as instituições que mais geraram comportamentos anômalos dentro do espaço de tempo analisado, para as quais dá-se o nome de influenciadores. A Figura 4.12 mostra esses dados para quando as instituições estão recebendo requisições de acesso. Já a figura 4.13 traz a análise de quando as instituições estão realizando requisições.

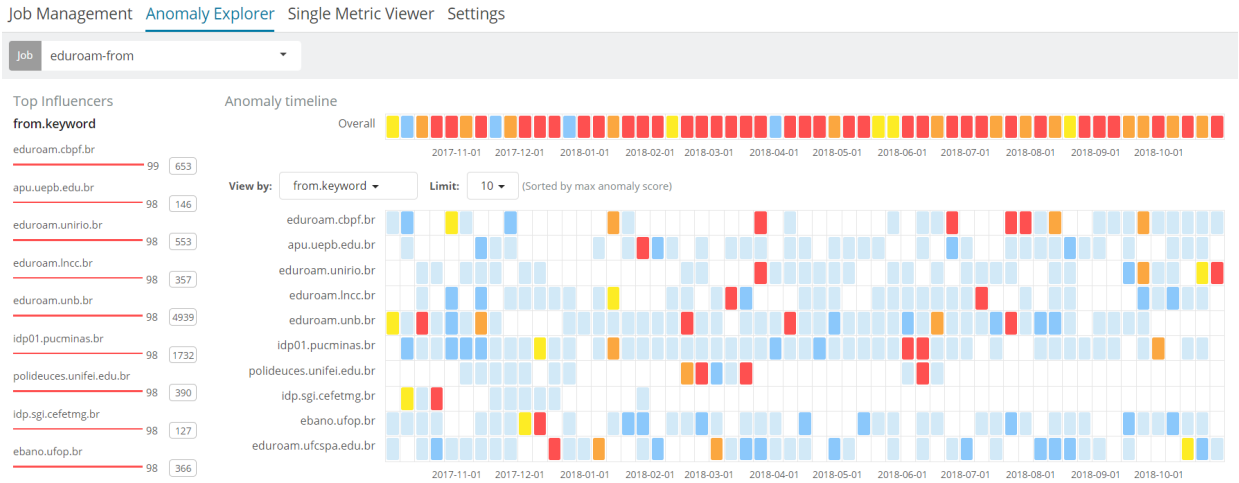


Figura 4.12: Visão geral das requisições recebidas por instituições no *Anomaly Explorer*.

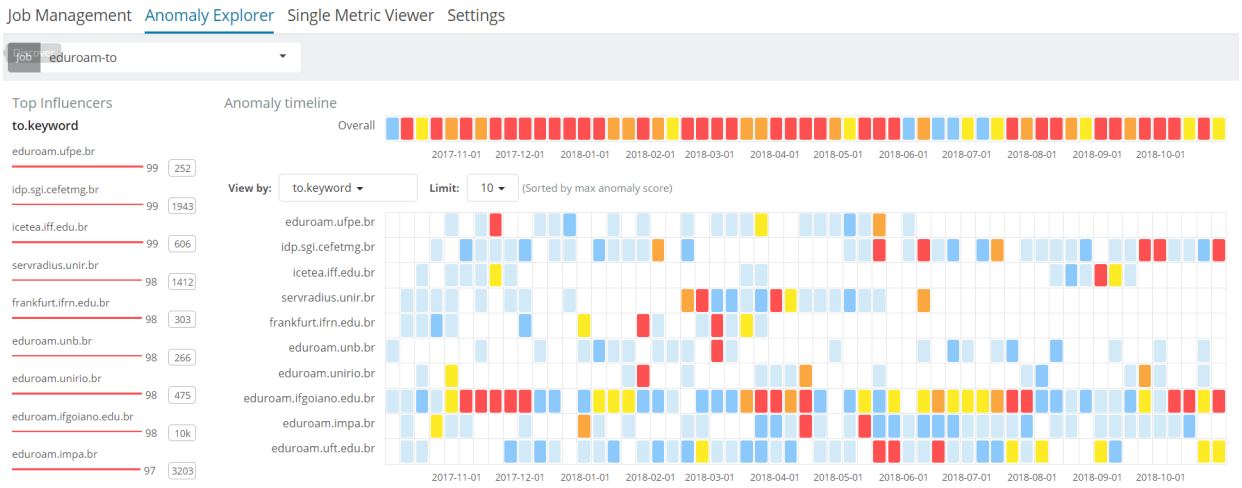


Figura 4.13: Visão geral das requisições realizadas por instituições no *Anomaly Explorer*.

A segunda forma de vermos os dados gerados pelas análises comportamental é denominada de *Single Metric Viewer*, ou Visualização de Uma Métrica. Essa é uma forma de visualizar, dentro de uma linha do tempo, onde estão as anomalias detectadas. As Figuras 4.14 e 4.15 descrevem essa visualização, para instituições recebendo e realizando requisições, respectivamente.

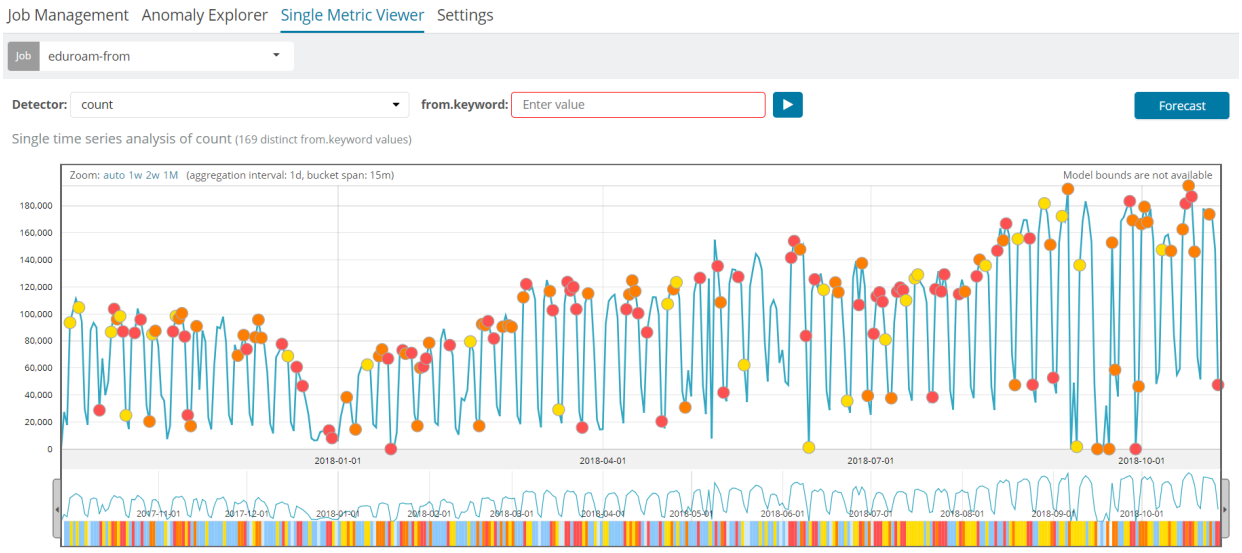


Figura 4.14: Visão geral das requisições recebidas por instituições no *Single Metric Viewer*.

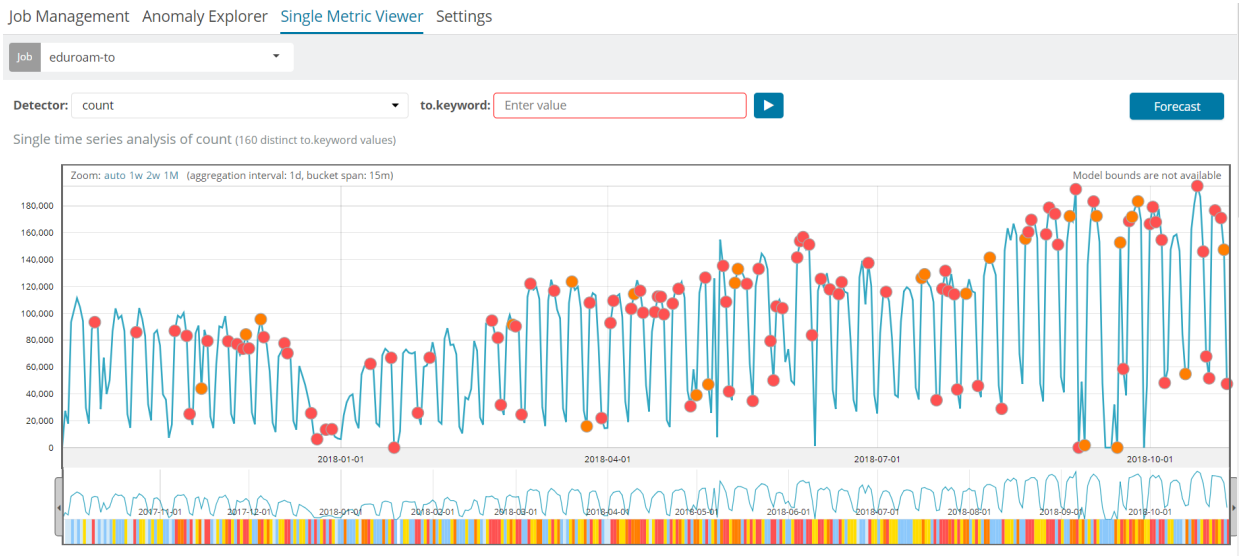


Figura 4.15: Visão geral das requisições realizadas por instituições no *Single Metric Viewer*.

4.2.2 Análise Individual

A partir da análise vista na seção anterior, podemos refinar nossa busca e análise para instituições em específicos. Se observarmos a página mostrada, podemos ver um campo "from.keyword"ou "to.keyword", dependendo se a análise foi feita para instituições recebendo ou realizando requisições, respectivamente. Nesse campo, podemos selecionar a instituição desejada para verificar a análise para aquela instituição.

Para efeito de exemplificação, foram escolhidas, aleatóriamente, uma instituição para a análise de requisições realizadas e uma para as recebidas.

A Figura 4.16 mostra o desempenho do Insituto Federal de Minas Gerais realizando requisições. O que podemos observar nessa análise, é que o IFMG ficou um longo período do ano com pouquissímas requisições realizadas. Porém, a partir do mês de julho, essa utilização aumentou. Vemos que a ferramenta identificou que, no início, essa grande utilização era uma anomalia, mas depois ela se adaptou. O mesmo acontece quando essa utilização cai e depois volta a subir. É possível, diante dessa amostra, aferir que o IFMG teve um período de inatividade entre o final de agosto e início de outubro. Esse problema poderia ter sido evitado caso essa ferramenta de monitoramento já estivesse sendo utilizada.

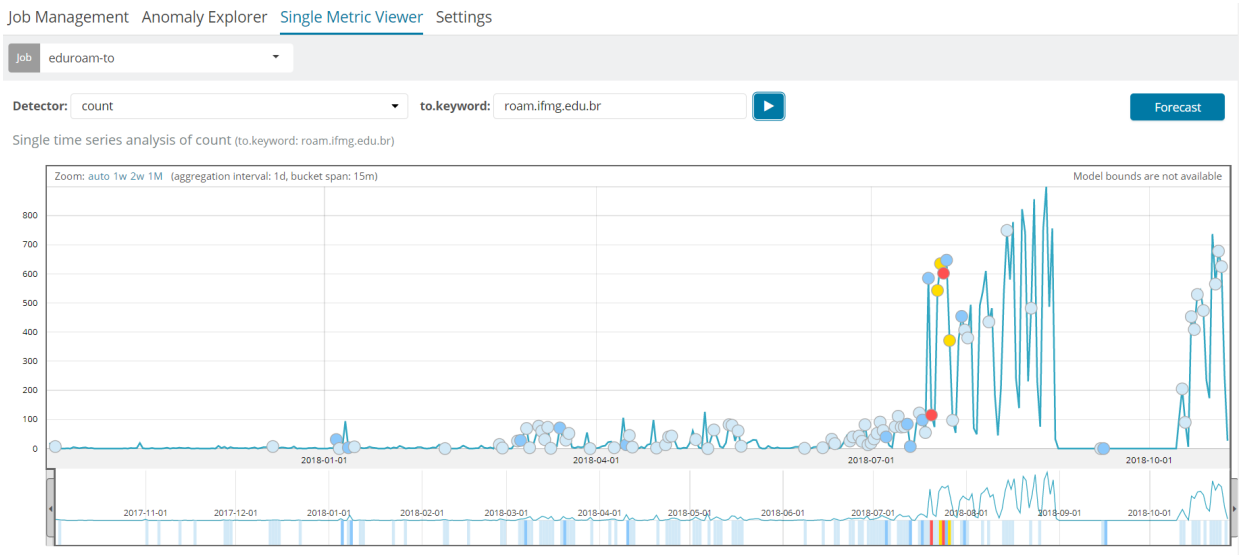


Figura 4.16: Análise individual da utilização do eduroam do IFMG realizando requisições.

Já na Figura 4.20, observamos o comportamento da PUCMINAS recebendo requisições durante o ano de 2018. Vemos que manteve-se uma média de utilização durante o ano, com alguns picos para cima e outros para baixo. O importante é que, nos picos negativos, não houve um grande período de inatividade. Portanto, a PUCMINAS é uma instituição que não encarou problemas na utilização do eduroam em 2018, ao menos quando estava enviando usuários para outras instituições.

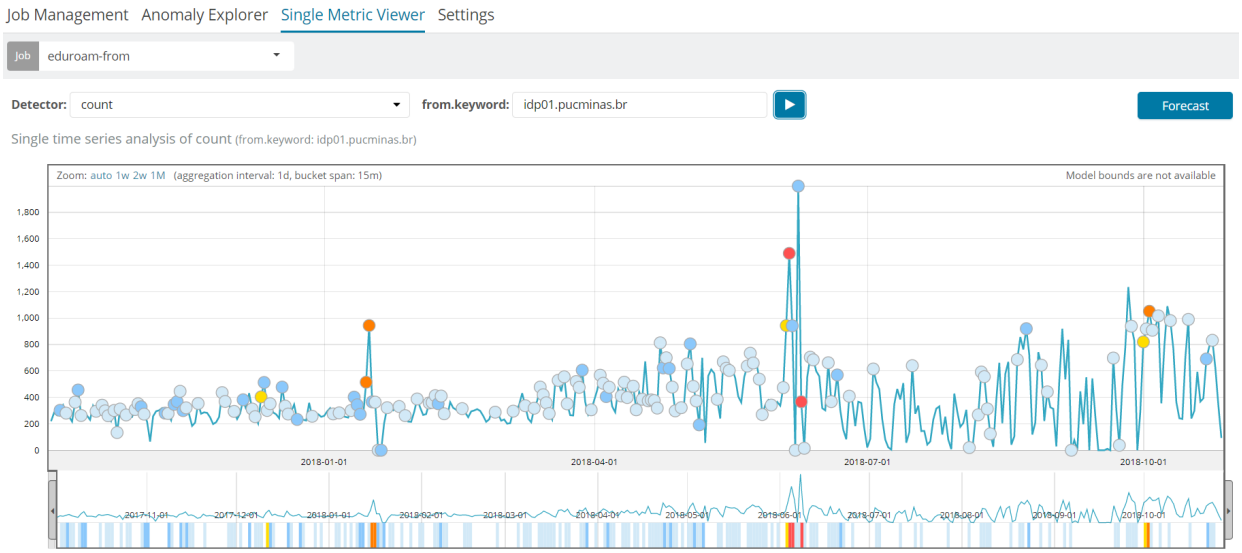


Figura 4.17: Análise individual da utilização do eduroam na PUCMINAS recebendo requisições.

4.3 Alertas Gerados

A partir dos dados geranos com a análise comportamental, demonstrada na seção anterior, são criados diferentes alertas, enviados por e-mail, para que o monitoramento do eduroam atue de

forma rápida para resolver possíveis problemas identifados. De acordo com a necessidade observada para o monitoramento, foram criados três tipos de alerta, que serão detalhados a seguir.

O intuito desse procedimento é a abertura de chamados diretamente no *Service Desk* da RNP. Porém, para a realização dos testes foi usado um e-mail pessoal para envio e recebimento desses alertas gerados.

4.3.1 Indisponibilidade do serviço

O primeiro alerta criado para agir no monitoramento do eduroam é o indisponibilidade do serviço como um todo. Este alerta é acionado quando o servidor *radsecproxy* da RNP fica mais de dez minutos sem receber nenhuma requisição. Isso indica que há uma possível insdisponibilidade no serviço e, por isso, é gerado o alerta.

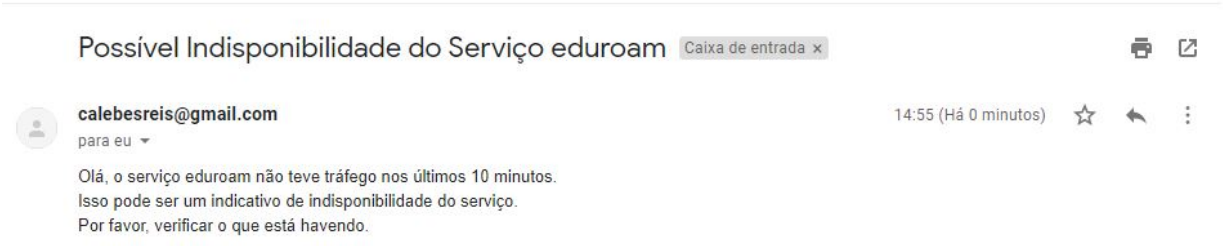


Figura 4.18: Indisponibilidade geral no eduroam.

4.3.2 Instituições sem Receber Requisições

Os outros dois alertas criados são de análises individualizadas. O primeiro deles verifica quais instituições estão há um determinado período sem receber requisições. Esse período ainda está sendo testado, para determinarmos um tempo preciso para essa análise. Vale ressaltar que a utilização do *Machine Learning* nos ajuda a diminuir os erros nessa análise, pois instituições que não fazem ou recebem requisições normalmente não geram anomalias.

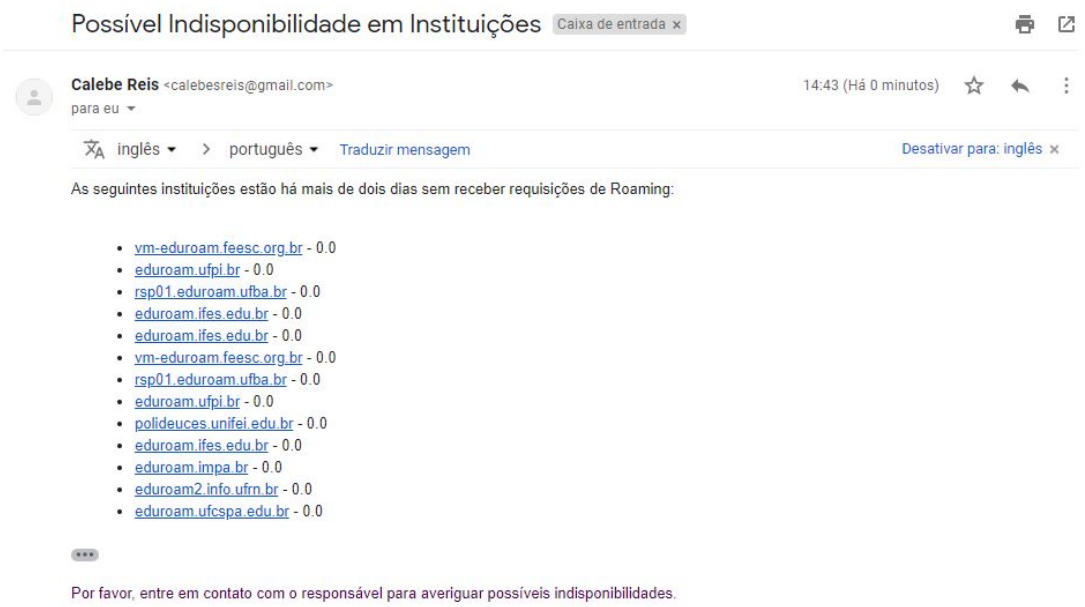


Figura 4.19: Indisponibilidade no recebimento de requisições.

4.3.3 Intituições sem Fazer Requisições

O outro alerta individualizado é sobre as instituições que estão sem fazer requisições. Assim como no anteior, o melhor período para essa análise ainda não está determinado.

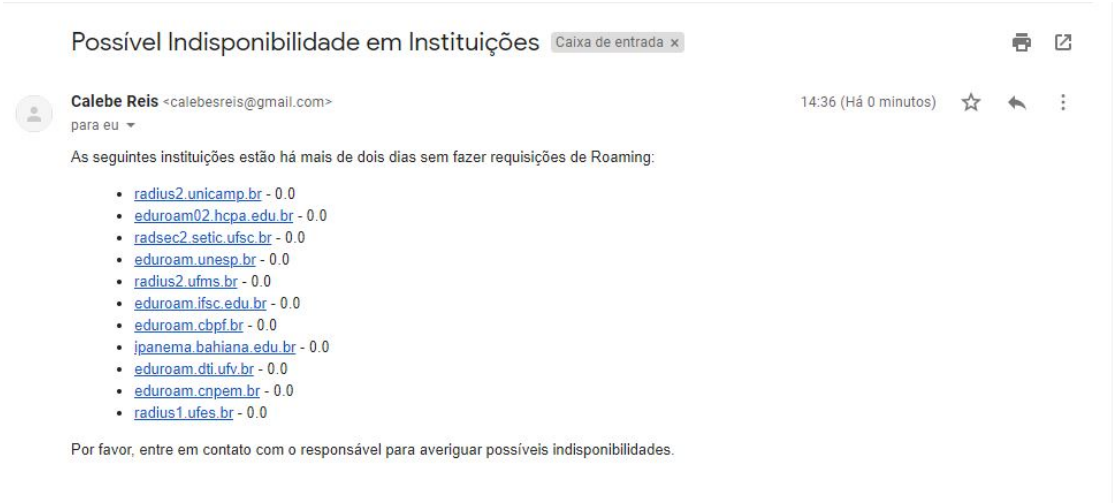


Figura 4.20: Indisponibilidade na realização de requisições.

Capítulo 5

Conclusão

Neste projeto foi implementada uma solução de monitoramento do processo de *roaming* do serviço eduroam. A prova de conceito realizada com essa implementação tinha o intuito de evidenciar situações, até então desconhecidas, em que instituições estavam tendo problemas com a utilização desse serviço. A ferramenta utilizada para fazer esse monitoramento também é útil para gerar estatísticas de uso do eduroam, nos possibilitando criar visualizações e dashboards para melhor entendimento dos dados coletados. O trabalho foi realizado em parceria com a RNP, envolvendo pessoas de equipes diferentes, o que prova a autenticidade dos dados coletados e das análises realizadas. A gerência do serviço eduroam na RNP aprovou o trabalho realizado, autorizando, assim, a implementação desse monitoramento no ambiente de produção do serviço. Portanto, podemos concluir que o trabalho cumpriu com o objetivo proposto de trazer à realidade uma forma de melhor analisar e monitorar a utilização do eduroam nas instituições brasileiras de ensino e pesquisa.

5.1 Trabalhos Futuros

A proposta de monitoramento apresentada neste trabalho é apenas o início de uma melhoria do serviço eduroam. Para continuar esse processo de melhora, foram pensadas algumas medidas que podem ser tomadas para que o monitoramento seja cada vez mais preciso. A primeira nova implementação que pode ser feita é a inserção de Georreferenciamento à plataforma, tendo em vista que a ferramenta tem diversas opções para tratar dados georreferenciados. Outros dois pontos a serem realizados no futuro são de nível de colaboração com outras instituições. Primeiramente, deseja-se mostrar essa solução para as instituições para que cada instituição possa monitorar o seu servidor *radsecproxy* interno. A outra medida que pode ser tomada é a apresentação desse projeto de monitoramento para a Confederação Latino Americana, para que também seja pensada uma solução para monitorar o servidor da Confederação.

Bibliografia

GÉANT. **Eduroam is now available in 101 countries**. 2018. Disponível em: <https://www.eduroam.org/2018/11/07/eduroam-now-available-in-101-countries/>. Acesso em: 20 nov. 2018.

IEEE. **IEEE 802.11-1999 - Standard for Information Technology - Telecommunications and information exchange between systems - Local and Metropolitan Area networks - Specific requirements - Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications**. [S.l.: s.n.], 1999. Disponível em: <http://standards.ieee.org/findstds/standard/802.11-1999.html>. Acesso em: 10 nov. 2018.

_____. **IEEE Standard for information technology - Telecommunications and information exchange between systems-Local and metropolitan area networks-Specific requirements-Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications: Amendment 6: Medium Access Control (MAC) Security Enhancements**. [S.l.]: IEEE, jul. 2004. DOI: 10.1109/ieeestd.2004.94585. Disponível em: <https://doi.org/10.1109/ieeestd.2004.94585>. Acesso em: 10 nov. 2018.

IETF, Internet Engineering Task Force. **Remote Authentication Dial In User Service (RADIUS)**. 1997. Disponível em: <https://www.ietf.org/rfc/rfc2058.txt>. Acesso em: 15 nov. 2018.

IFSC. **Estudo e implementação de uma infraestrutura eduroam**. 2012. Disponível em: <https://wiki.sj.ifsc.edu.br/wiki/index.php/Estudo_e_implementa%5C%C3%5C%A7%5C%C3%5C%A3o_de_uma_infraestrutura_eduroam>. Acesso em: 21 nov. 2018.

KAMMERSTETTER, Markus et al. Efficient High-Speed WPA2 Brute Force Attacks using Scalable Low-Cost FPGA Clustering [Extended Version]. **CoRR**, abs/1605.07819, 2016. arXiv: 1605.07819. Disponível em: <http://arxiv.org/abs/1605.07819>.

RNP, Rede Nacional de Ensino e Pesquisa. **CAFe - Comunidade Acadêmica Federada**. 2018. Disponível em: <https://www.rnp.br/servicos/servicos-avancados/cafe>. Acesso em: 15 nov. 2018.

SAADE, Débora c. **Eduroam: Acesso sem fio seguro para Comunidade Acadêmica Federada**. Escola Superior de Redes, 2013.

SAS. **Machine Learning, Qual a sua importância?** 2018. Disponível em: <https://www.sas.com/pt_br/insights/analytics/machine-learning.html>. Acesso em: 13 nov. 2018.

ANEXOS

ANEXO A

Instalação e configuração das ferramentas

A.1 Instalação e configuração do Kibana

Inicialmente, é necessário instalar o JAVA.

```
$ sudo apt install openjdk-8-jre-headless
```

Para então, instalarmos o Kibana

```
$ sudo wget https://artifacts.elastic.co/downloads/kibana
/kibana-6.4.2-amd64.deb
```

```
$ sudo dpkg --i kibana-6.4.2-amd64.deb
```

Depois de instalar, é preciso configurar o Kibana

```
$ sudo vim /etc/kibana/kibana.yml
```

server.host: "0.0.0.0"
elasticsearch.url: "http://(200.139.35.11:9200)"

Para que o Kibana responda na porta 80, é necessário instalar o Nginx como proxy reverso

```
$ sudo apt install nginx apache2-utils
```

E configurar login e senha para acesso à plataforma

```
$ sudo htpasswd -c /etc/nginx/htpasswd.users kibanaadmin
```

Devemos também configurar o Nginx

```
$ sudo unlink /etc/nginx/sites-enabled/default
```

```
$ sudo vim /etc/nginx/sites-available/kibana
```

```
server
listen 80;
server name (200.139.35.26);
auth basic "Restricted Access";
auth basic user file /etc/nginx/htpasswd.users;
location / proxy pass http://localhost:5601;
proxy http version 1.1;
proxy set header Upgrade $http upgrade;
proxy set header Connection 'upgrade';
proxy set header Host $host;
proxy cache bypass $http upgrade;
```

Por fim, criamos o link simbólico

```
$ sudo ln -s /etc/nginx/sites-available/kibana /etc
/nginx/sites-enabled/kibana
```

A.2 Instalação e configuração do Elasticsearch

Para instalar o Elasticsearch,

```
$ sudo wget https://artifacts.elastic.co/downloads
/elasticsearch/elasticsearch-6.4.2.deb
```

```
$ sudo dpkg --i elasticsearch-6.4.2.deb
```

Para configurar o Elasticsearch,

```
$ sudo vim /etc/elasticsearch/elasticsearch.yml
```

```
network.host: "0.0.0.0"
```

A.3 Instalação e configuração do Logstash

Inicialmente, é necessário instalar o JAVA.

```
$ sudo apt install openjdk-8-jre-headless
```

E então, instalar o Logstash,

```
$ sudo wget https://artifacts.elastic.co/downloads
/logstash/logstash-6.4.2.deb
```

```
$ sudo dpkg --i logstash-6.4.2.deb
```

Para configurar o Logtash, é necessário criar o seguinte arquivo

```
$ sudo vim /etc/logstash/conf.d/lohstash.conf
```

```
input
beats
port => 5044
type => syslog

filter
grok
match => "message"=>
"%MONTH:mes*%MONTHDAY:dia*%TIME:hora*%YEAR:ano:.
*from%HOSTNAME:fromto%HOSTNAME:to%IP:ip$
mutate  add field =>  "timedate"=> "%mes %dia %hora

%ano
output
elasticsearch
hosts => ["200.139.35.11:9200"]
```

A.4 Instalação e configuração do Filebeat

Para instalar,

```
$ sudo wget https://artifacts.elastic.co/downloads
/beats/filebeat/filebeat-6.4.2-amd64.deb
```

```
$ sudo dpkg --i filebeat-6.4.2-amd64.deb
```

Para configurar,

```
$ sudo vim /etc/filebeat/filebeat.yml
```

```
paths:
-/var/log/radsecproxy/radsecproxy.log
output.logstash:
Hosts: ["(200.139.35.25):5044"]
```

ANEXO B

Arquivos para criação de alertas

B.1 Instituições sem realizar requisições

```
{
  "trigger": {
    "schedule": {
      "interval": "15m"
    }
  },
  "input": {
    "search": {
      "request": {
        "search_type": "query_then_fetch",
        "indices": [
          ".ml-anomalies-eduroam-to"
        ],
        "types": [],
        "body": {
          "size": 100,
          "query": {
            "bool": {
              "must": [
                {
                "script": {
                "script": {
              "source": "doc['actual'].value < doc['typical'].value",
              "lang": "painless"
                    }
                  }
                },
```

```
                {
                  "range": {
                    "timestamp": {
                      "gte": "now-1y"
                    }
                  }
                }
              ]
            }
          }
        }
      }
    }
  },
  "condition": {
    "always": {}
  },
  "actions": {
    "send_email": {
      "email": {
        "profile": "standard",
        "to": [
          "rnpeducras@gmail.com"
        ],
        "subject": "Possivel indisponibilidade em instituicoes",
        "body": {
          "html": "As seguintes instituicoes estao ha mais de um dia
          sem realizar requisicoes de Roaming <ul>Encontrados
          {{ctx.payload.hits.total}}</ul> <ul>{{#ctx.payload.hits.hits}}
          <li>{{_source.partition_field_value}} -
          {{_source.actual.0}}</li>{{/ctx.payload.hits.hits}}</ul>"
        }
      }
    }
  }
}
```

B.2 Instituições sem receber requisições

```
{
  "trigger": {
```

```
  "schedule": {
    "interval": "15m"
  }
},
"input": {
  "search": {
    "request": {
      "search_type": "query_then_fetch",
      "indices": [
        ".ml-anomalies-eduroam-from"
      ],
      "types": [],
      "body": {
        "size": 100,
        "query": {
          "bool": {
            "must": [
              {
              "script": {
              "script": {
            "source": "doc['actual'].value < doc['typical'].value",
            "lang": "painless"
                  }
                }
              },
              {
                "range": {
                  "timestamp": {
                    "gte": "now-48h"
                  }
                }
              }
            ]
          }
        }
      }
    }
  }
},
"condition": {
  "always": {}
},
```

```
  "actions": {
    "send_email": {
      "email": {
        "profile": "standard",
        "to": [
          "rnpeducras@gmail.com"
        ],
        "subject": "Possivel indisponibilidade em instituicoes",
        "body": {
          "html": "As seguintes instituicoes estao ha mais de um dia
          sem receber requisicoes de Roaming <ul>Encontrados
          {{ctx.payload.hits.total}}</ul> <ul>{{#ctx.payload.hits.hits}}
          <li>{{_source.partition_field_value}} -
          {{_source.actual.0}}</li>{{/ctx.payload.hits.hits}}</ul>"
        }
      }
    }
  }
}
```